Kalyani Penubarthi

B2B APIs – Supply Chain Collaboration

Helsinki Metropolia University of Applied Sciences

Master of Engineering

Information Technology

Master’s Thesis

22 November 2018



PREFACE

Completing this thesis formed a major milestone and culmination of my long journey of interaction with business managers, solution architects, project managers, plant managers and strategic buyers. It has continuously forced me to zoom-in for the details and zoom-out to get a strategic perspective. During the process, multitude of ideas are extruded through renewed frameworks that produced a practical implementable view of APIs. I am sincerely thankful to all my colleagues from Finland, Italy, Netherlands, and I would like to express my gratitude to the professors at Metropolia, my course coordinator Ville Jääskeläinen and my thesis guide Auvo Häkkinen.

Finally, a big thanks to my kids – Rythama and Laasya, who unconditionally extended their love and affection and supported me throughout these extremely long working years.

Helsinki, 22 November 2018

Kalyani Penubarthi



Author(s) Title Number of Pages Date	Kalyani Penubarthi B2B APIs – Supply Chain Collaboration 57 pages 22 November 2018
Degree	Master of Engineering
Degree Programme	Information Technology
Specialisation option	Networking and Services
Principal Lecturers	Auvo Häkkinen Ville Jääskeläinen

Abstract

In the programming world, APIs (Application Program Interfaces) exist in different forms since the early ages of software development. However, they were built in silos with limited scope. With rapid advance of digitalization, the possibilities are advanced from mere IT integration initiative to Business Mandate – A mandate to exchange transaction data and application logic amongst the external value chain partners. This thesis focuses on usage of APIs in Supply Chain domain. This study is conducted with an anonymous (name is withheld) manufacturing organization. This study analyses the existing modes of integration, their challenges and how APIs can give a strategic advantage. The intent of the organization is to add APIs to the strategy in order to replace or compliment the current B2B integration methods. The research continues on finding what it takes an organization to introduce APIs to their Digital collaboration tool kit as a new integration method. This study highlights the preparation that need to be done not only within the enterprise, but also with the value chain partners, and the related change management. In addition, the research presents the detailed survey conducted with already existing B2B integration partners of this anonymous organization to assess their maturity. While some partners are in the initial stages of awareness, some are knowledgeable and willing to travel towards transformative API paradigm.

Keywords	API, B2B, Supply Chain Collaboration, ERP



Table of Contents




Helsinki
Metropolia
University of Applied Sciences

List of Figures



List of Abbreviations

APAC	Asia Pacific
ANSI	American National Standards Institute
API	Application Program Interface
ASN	Advance Shipment Notification
B2B	Business-to-Business
CPG	Consumer Package Goods
CORBA	Common Object Request Broker Architecture
ebXML	Electronic Business using eXtended Mark-up Language
EDI	Electronic Data Interchange
EDIFACT	Electronic Data Interface for Administration, Commerce and Transport
EJB	Enterprise Java Beans
EMEA	Europe Middle East and Africa
ERP	Enterprise Resource Planning
ESB	Enterprise Service Bus
IIOP	Internet Inter Operable Protocol
JRMP	Java Remote Method Protocol
MRP	Material Requirement Planning
MRP II	Manufacturing Resource Planning
ORB	Object Request Broker
P2P	Purchase-to-Pay
PO	Purchase Order
REST	REpresentational State Transfer
RFQ	Request for Quotation
RMI	Remote Method Invocation
RPC	Remote Procedure Call
SCM	Supply Chain Management
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SWIFT	Society for Worldwide Interbank Financial Telecommunication
UN/ EDIFACT	United Nations / EDIFACT (described above)
XML	eXtended Mark-up Language



1 Introduction

API (Application Programming interfaces) is not a new phenomenon and has been around from the times of beginning of Software applications but, with the rapid changes towards digitalization in the business collaboration methods, the topic of Business or transactional APIs is being widely discussed now-a-days. APIs are unsung heroes of the digital age – as published in the Business Matters magazine's article [14]. To understand APIs, think of them as messengers who takes requests and, when tell them what to do will return the response back says software firm Mulesoft. Businesses of size large or small can benefit by improving their operations efficiency by collaborating in a better way with their supplier/customers/partners with APIs. In this thesis paper we are taking the scenario of a multinational manufacturing organization regarding their supplier collaboration ecosystem with transactional APIs.

A Transactional/business API is a Persona for an enterprise, exposing defined data and Services for Partner consumption. This will allow a manufacturing organization to open the new markets or increase operations efficiency by allowing partner companies for improved collaboration, achieving greater transparency and visibility in supply chain, identifying and minimizing risk, eliminating errors in data, enable more accurate forecasting. There can be several APIs which can be developed and published for suppliers depending on the manufacturing industry. In principle, most of the available connections in Electronic Data Interchange (EDI) can also be developed as APIs. APIs could be for Request for Quotation, Purchase Order, Order Confirmation, Billing, Payment, Transportation booking, and Certificate of Origin, Collaborative Planning, Forecasting, and Replenishment.

This study is a qualitative analysis and will build an open thinking and investigate what kind of possibilities and challenges would be there in a global scale manufacturing organization to introduce, this concept of APIs in to their strategy as a start and also to develop, and implement to their supplier ecosystems.



1.1 Research Objectives

The objective of this study is to investigate the potential of APIs from the publishing manufacturing organization’s perspective. The study will especially focus on collaboration potential of APIs with suppliers, benefits and challenges faced in this total supply chain eco system when respective parties are trying to develop, implement and maintain the APIs. Also, if possible, collect a survey data about the readiness and maturity level of the Suppliers towards the change of embracing APIs.

Due to the sparseness in the earlier research in this very area of consideration and also APIs being still fairly the young and less explored collaboration method, the research also makes use of selected online resources and Supply chain related online material to build the understanding of the issue. The study is exploratory in nature and aims to find some promising prospects for further research.

1.2 Research Questions and Methods

The empirical part of the study is qualitative in nature and aims to provide detailed perspective and challenges for organizations who would like to introduce APIs in to their supplier collaboration toolkit. The thesis uses case study approach to focus on an anonymous global manufacturing industry. A structured discussion within the organization with the IT department and also with different business stakeholders along with the study of various online sources concludes to primary data collection method for the study to understand the maturity for this ecosystem. This method provides flexibility, which is required when topic of the research is fairly new and a little studied phenomenon such as Supplier collaboration APIs. In addition, a survey has been carried out with the partner ecosystem to understand the awareness and maturity of this supply chain collaboration concept.

Main research questions of the study are the following:

1. What is the role of APIs in B2B collaboration for manufacturing industries?

The question is on the study of the topic based on the interviews held in the anonymous organization and the material available online. This is explained in detail in chapter 2.

2. What are the benefits of these collaboration APIs for both provider (manufacturing industry) and partners?

This questions aims to find out the overall benefits with respect to organization and its supplier ecosystem and how these stakeholders can take advantage of this opportunity.



Potential benefits of API will be investigated closer in the literary review section of the study.

3. What are the key Challenges?

This question revolves around the scenario from including APIs into the collaboration strategy to development, implementation/promotion to suppliers and also the possible challenges in the maintenance of the APIs. Challenges are discussed to some level in Chapter 2 and in detail in chapter 6.

1.3 Structure of Thesis

First chapter gives the introduction to thesis, objectives and the structure of the document.

Chapter 2 contains the concept of business to business integrations. This also contains information regarding the current supplier collaboration platforms (B2B) and its evolution. In this chapter, I tried to analyze about how this is moving towards the future and also about the API way of collaboration in manufacturing industries. In this chapter, it is also compared with the currently available technologies in the pretext of differentiation (Pros and Cons). This chapter would give a fair idea about the industry and the thesis topic study and their status along with the benefits and the possible challenges.

API literature has been included in the chapter 3 which gives a detailed view about the Web APIs used for B2B integrations. In this chapter, it is also been analyzed about the challenges regarding standards as this has been a rather new way of B2B integrations.

Chapter 4 presents the research made in an anonymous manufacturing organization and gives the analysis about what it takes an organization to develop API strategy and include it to the supplier collaboration toolkit starting from the internal analysis to the readiness in the partner ecosystems. Challenges faced by the industry are also presented in detail in this chapter.

A partner awareness and readiness survey has been conducted through the anonymous organization in order to better understand the situation of the Supplier ecosystem to this new technology. Chapter 5 contains the analysis of the survey.

Chapter 6 concludes the thesis with research findings and recommendations.

2 Analysis of Business to Business integrations

In this chapter is explained the evolution of business collaborations keeping the perspective of manufacturing organization. Starting from development of ERP for the internal usage to store and manage the data and transactions in an organized way, and then moving towards the collaboration technologies. This chapter also presents the growing importance of B2B integrations and the budgetary trends in organizations from different industries referring to the Stanford study. In continuation, the main B2B integration technology EDI has been presented along with the benefits it fetched and the challenges towards the future. Finally the analysis is taken forward by the introduction of APIs for B2B integration concept. The API literature itself has been presented in the next chapter.

2.1 Enterprise Resource Planning

In the initial times, the concept was applied to inventory management in manufacturing industries. Software programs are created for monitoring inventories, reconcile the same with physical inventories and create a report in areas where differences are found. In the next decade, these are evolved into adding the Material Requirement Planning processes that consider the demand and Bill of Materials. These programs started gaining the traction in Production Planning areas.

By 1980s, typical Material Requirement Planning evolved further to consider the other resources involved in the manufacturing planning. These are called as Manufacturing Resource Planning for MRP – II. With the gaining popularity and acceptance, these software programs evolved further. By mid 1990s, other functionalities like human resources and finance are added to these. This is evolved into a suite of products that cater the needs of multiple part of enterprise, and then the ERP has become much more widespread. Many commercial software vendors started offering these programs. [1]

ERP symbolizes enterprise resource planning. It is a combination of software systems used by organizations to run essential business functions, such Planning, Procurement, Inventory, Accounting, Human resources, Customer relationship management etc., ERPs combine the transactional data together with processes and define a system of business processes which facilitates data flow across them. By collecting an organization’s shared transactional data from multiple sources, ERP systems eliminate



data duplication and provide data integrity with a “single source of truth.” Today, ERP systems are critical to manage small, medium and enterprise businesses across all industries.

ERP brings the several business values to the Organizations like: improvement in business insight, less operational costs, enhanced collaboration, improved efficiency, reduced risk, consistent infrastructure, high user-adoption rates, lower management and operational costs.

With the increasing complexity of the business models and processes, it is extremely necessary for the organizations to collaborate and share the information with the partners in order to manage the day-to-day business. In this study, we are considering the Supply Management area for B2B integrations.

2.2 Introduction and Importance of B2B Integrations

As cited in OpenText whitepapers, businesses are no longer successful by solely operating within their own four walls [2]. A modern corporation is critically dependent upon a network of business partners and their ecosystem to function. Consider a typical supply chain in a manufacturing organization: Procurement of the materials can be made from geographies where the costs are economical and also depending on the availability of those. Freight carriers, logistics and, transportation carriers move the goods to consolidation points or warehouses. Operational functions are increasingly becoming complex depending upon external entities as the outsourcing trend continues to grow. To operate with today’s highly global, distributed and complex business models, it requires real-time connectivity with the network of business partners in order to be successful. Without visibility into the locations of inventory, the forecasts of demand, the availability of supply and the status of payments, companies cannot make decisions related to day-to-day operations.

In spite for this high demand and need for business-to-business connectivity, the interactions between organizations and their partners is not yet very efficient. Large percentage of the information exchanged between business partners travels over email, phone and fax rather than flowing directly between business applications via B2B integration technologies. Integration touch points across partners is represented in Fig 1.

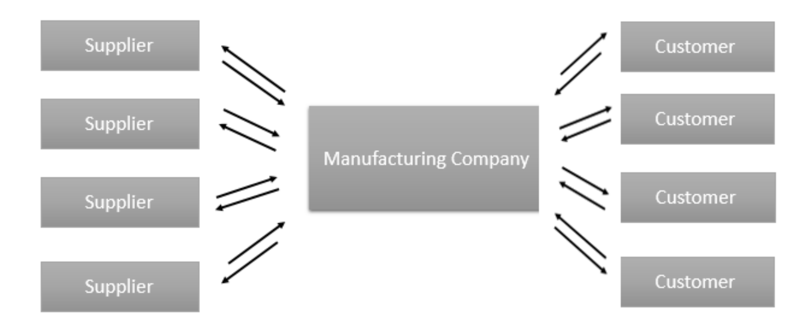


Fig 1 – B2B Integrations

In the past decade, publishing transactional information to Web Portals and providing secure access to the partners has seen major light. This has become as one of the best choices for bigger companies. This might not still work effectively if the volumes of the transactions is more as it will lead to lot of manual work at the other end.

According to Stanford Business [16], despite the number of B2B technologies available, these are not very well adopted in business community. It is not due to the shortage of options and approaches and methods for connectivity. There exist many offering in B2B connectivity that can support wide variety of businesses different degrees of volumes. But the number of custom developments built amongst the business partners and the change management involved when there comes new acquisition or change in business context, makes the connectivity much difficult.

With globalization new doors are opened for business communities to find supply sources which are economical and efficient. From the P&L perspective, it is great benefit. At the same time, it results in complex supply and distribution networks. Considering the different time zones, languages, volumes, the traditional methods of integration no longer stay relevant, and electronic way of collaboration amongst the trading partners becomes the key. Realizing this, an increased number of companies have taken steps to invest in electronic communication capabilities and B2B collaboration has become a reality. Some of the resulting benefits of such B2B integration programs include improved operational efficiencies, reduced costs, and increased Customer satisfaction. With new forms of technological innovations like Internet of Things, there a need for a step change in this



connectivity, and demand for B2B connectivity solutions is expected to increase in a faster pace than ever before.

With the fast paced development in B2B integration technologies currently, and the growing interest in B2B managed services, the Stanford Global Supply Chain Management Forum in collaboration with GXS, initiated a research study in 2012 [16]. The research information is summarized here to get the understanding of how much of importance is the B2B collaborations for divergent industries with different sizes. This would give the depth of the topic here in discussion. The goal of this study was to gain insights into the latest trends and perceived business value associated with B2B integration technologies and B2B managed services. The study was based on a survey distributed to current users of B2B integration technologies and B2B managed services.

In total, 92 people from 75 companies completed the survey. The primary industry of the participating companies varied (CPG4, financial services, logistics, manufacturing, retail, and other) as well as their annual revenue (from under $100 million to more than $5 billion),and their geographic base (North America, EMEA, and APAC). The following are some insights that emerge from analyzing the data provided by the participants regarding their B2B integration programs:

Budgetary trends: Half of the companies had an annual budget for B2B integration technologies of $1 million or more; often times—but not always—smaller companies tend to have smaller B2B budgets. For most companies (66 percent), their B2B integration budget represented 5 percent or less of their total IT budget. A larger portion of the small- and medium-sized companies in this group spent less than 1 percent of their IT budget on B2B, while a larger portion of very large companies spent 1 to 5 percent of their IT budget on B2B. The majority of companies (68 percent) saw their B2B budget increasing over the past three years, often by up-to 10 percent. Sixty-two percent of companies expected their B2B budget to Increase over the next three years, with most of them estimating an increase of up to 10 percent. Very few companies (7 percent) reduced their B2B budget over the past three years or expected it to decline in the near future. When considering the geographic location of participants, EMEA-based companies tended to have a larger annual budget dedicated to B2B integration technologies, and on average their B2Bbudget represented a larger portion of their total IT budget. In addition, a larger portion of them increased their B2B budget over the past three years and/or anticipated their budget to increase over the next three years. [16]



B2B transactions and connections: All but a very small fraction of participating companies expected their B2B transaction volume and number of connections to increase over the next three years, with the majority of them expecting an increase of up to 25 percent in B2B transactions and number of connections (54 and 65 percent of companies, respectively). Close to half of participating companies (40-50 percent) currently exchange transactions electronically with less than 20 percent of their customers, suppliers, or other business partners. At the other end of the spectrum, 7-10 percent of companies exchange transactions electronically with 81-100 percent of their customers, suppliers or other business partners. Companies often reported different strategies for each type of trading partner, with only 28 percent of them exchanging transactions electronically with the same proportion of their customers, suppliers and other partners. Most companies used a combination of structured messages (such as EDI, XML or SWIFT) and web portals for their B2B transactions. [16]

Process efficiency improvements: The majority of companies for which these questions were relevant saw an improvement in process efficiency as measured by the processing costs of orders and invoices received electronically compared to those received in other formats, as well as the costs to manage incoming shipments received at the companies' distribution centers, stores or manufacturing plants. The percent reduction in cost varied widely among the participating companies. [16]

Best-in-class companies: Best-in-class companies were defined as those that achieved more than 50 percent cost reduction in at least one of the potential process efficiencies listed in the survey. On average, these companies had a larger B2B budget, which also represented a larger portion of their entire IT budget. In addition, they expected a higher increase in B2B volumes and connections, and exchanged transactions electronically with a larger portion of most types of business partners. These companies also exchanged a higher portion of their transactions using structured messages. [16]

Future plans: When considering the companies' plans for future use of B2B e-commerce, 96 percent of participants plan to increase the number of customers they trade with electronically, the number of suppliers they trade with electronically, or the number of business processes they support. Of them, 59 percent plan to expand their use of B2B e-commerce in all three areas. [16]



Business-to-Business integration is rather an old concept and many organizations have been running this Integration projects with the help of their Information technology counter parts or partners since the late 1960's. Simply put, B2B integration means the integration, automation and optimization of key business processes that extend outside the four walls of an organization [3].

By connecting/integrating to externals meaning suppliers electronically, organizations can send the RFQs and get quotations, send purchase orders and receive the confirmation status, has the direct visibility for the shipment statuses, receive the traceability codes of the materials procured, get the quality certificates and receive invoices automatically to the ERP which would reduce the lot of manual work and in turn decrease the operational costs. This automation will also reduce the errors in the operational tasks and result in more data accuracy. And, on the other end, suppliers can process this order information faster and with less errors. Processing the orders in real-time by the suppliers allows them to be more responsible to their customers and hence can increase their service levels and in turn their sales.

B2B Integrations began with larger companies mandating methods of receiving business information through technology [3]. Slowly, it has evolved through the widespread adoption of Electronic Data Interchange (EDI) and in recent years has benefited from technology innovations e.g. the advent of the Internet, XML, web services and SOA, Business Process Management and SaaS. These innovations has led to the possibility of increased benefits available to companies of every size and model. As we explore in this Microsite there are a number of ways to implement B2B Integration solutions. We discuss that the solution approach should be driven by a company's business needs and objectives, rather than a particular implementation or technology set.

From the board room member's perspective, integration is a simple concept of bringing the different application written in different programming platforms or language, and different hardware components to work together and create an echo system where the information flows without getting into the nuances of underlying systems. As an example, as a customer service representative, while a Sales Order is entered, the expectation is to get the inventory visibility from the warehouse so that delivery date can be promised to the customer. It should happen irrespective of the platforms and technologies involved.



Though it sounds natural and fluent from the business user's perspective, teams involved in making such an integration possible, and technologies that facilitate these transactions are challenging depending on systems of choice.

During the process along with the software intellectual property vendors, Information Technology service providers have come up with multitude of solutions. Different automations are built. And then some of the workflows are productized to be industry specific. While the software intellectual property vendors came up products that could be configured and customized for different scenarios, service providers came up with productizing the workflow offerings. But most common principle in the background has been the EAI for both connectivity within an enterprise or across the enterprises. All major vendor like SAP, Oracle, Microsoft, IBM and many niche players have got solutions for different industries. B2B data which is transaction related could be for example, purchase Orders, invoices, dispatch/delivery notes, product/service details, inventory data, forecasts, etc.

2.3 Current Integration Technologies for B2B

With so much information flowing between organizations, there came the need to automate this to some extent. Consider the example of communicating a Purchase Order between two organizations. Although the PO is generated electronically in the ERP system of the organizations, but along the way, it might be needed to print into paper and manually send to or, through e-mail or fax the Supplier. On the other hand, Supplier's organization has to enter these details manually to their ERP systems. This adds up not only the manual work but also inconsistencies and error prone processes as the volumes tend to increase.

EDI – Electronic Data Interchange is a protocol maintained by the American national standards Institute used for the computer to computer exchange of business transactions. EDI allows organizations with their own business systems and ERPs to talk to each other using a standardized format. By automating the process, human interaction is minimized and the scope for delays and mistakes can be reduced and even eliminated. In the Fig 2 below typical document transferred are represented.

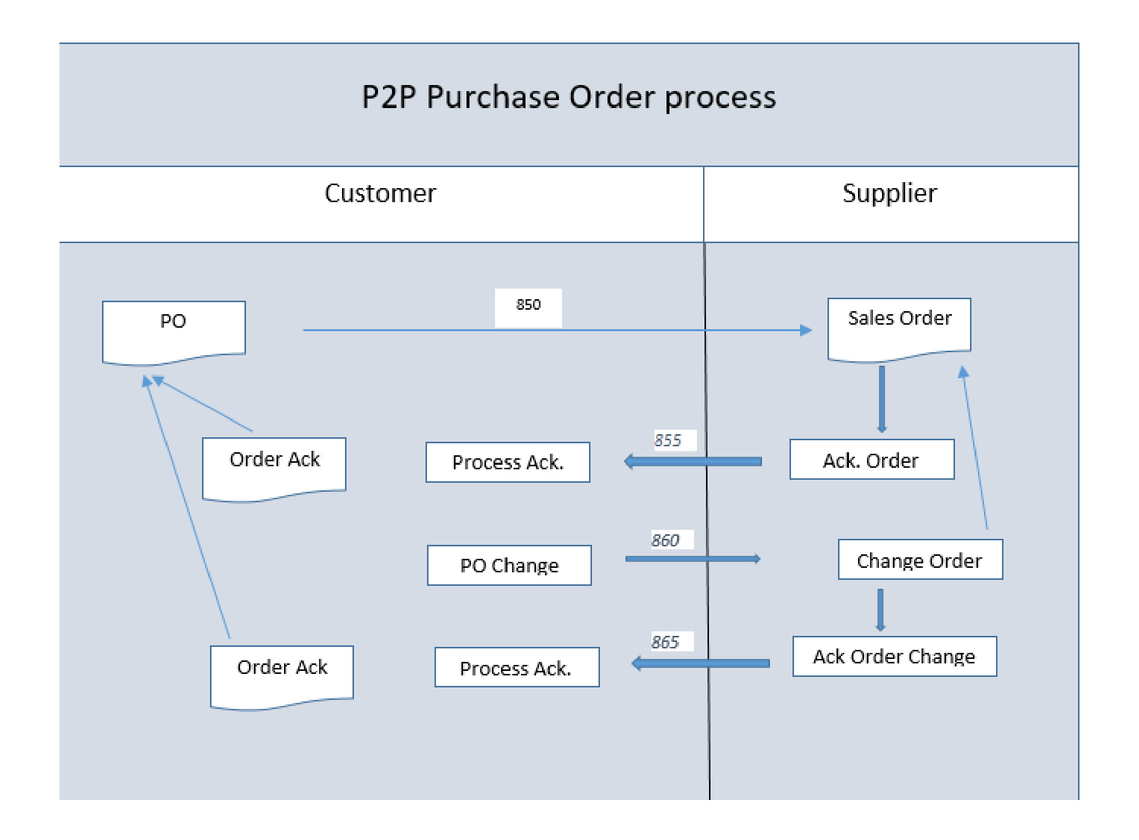


Fig 2 – Purchase Order Process

There are hundreds of documents that can be electronically transferred via EDI and the four most popular ones are:

Purchase Order - to initiate an Order

Invoice - sent to request payments

Advance Shipping Notification ASN - let a company buying products know when and how the items will be shipped. ASNs often include bar code information that indicate the specific items in each box.

Functional Acknowledgement – These are highly beneficial transactions that confirm the date and time a document transmitted via EDI was received. In this way, the sender will know the exact time and date when other party has received the information.

According to Forrester Research [4], EDI continues to prove its worth as an electronic message data format. This research states that “the annual volume of global EDI transactions exceeds 20 billion per year and is still growing.” Moving to EDI has numerous benefits to businesses like, improves business efficiency, improves transaction speed, improves visibility, improves document accuracy, reduces lead time, saves money, adds security and is environment friendly



2.3.1 EDI Standards

The technology behind EDI is 20-30 years old, and was designed to replace fax, postal mail, and email. Replacing human-to-human communication, whether through paper or electronic mail with automated EDI messages has been a cost-saver for many businesses. But because the EDI messages are automated, the messages must be in a standard format.

There are numerous EDI standards, including ebXML UN/EDIFACT, ANSI ASC X12, GS1 EDI, TRADACOMS, and HL7. For each standard, there are different versions, e.g., ANSI 5010 or EDIFACT version D12. This can create issues when businesses need to communicate with different EDI standards. The businesses will either have to agree on the EDI standard to use or employ some sort of translation service, whether in-house software or an EDI service provider. EDI standards prescribe both mandatory and optional information for any particular document and provide rules for the document's structure.

When an EDI document is created, such as an invoice, the order of text and the order of the data fields within that text gives it meaning. That text must strictly adhere to the EDI standard or else the sending and receiving systems will not be able to understand the document. This define exactly where and how each piece of information in the document will be found. An EDI message will comprise one single business document, e.g. a purchase order, invoice, or advance ship notice.

2.3.2 Components for Implementation

Across many industries, the exchange of electronic document B2B messages continues to be the means by which key business processes are transacted. To successfully implement a fully integrated EDI solution 4 components are required between 2 trading partners ie., manufacturing organization and the supplier who would like to exchange the information.

Mapping : They need to be able to automatically move transactions to and from their backend business software without having to re-type it. This is called integration. Data mapping software ensures that the EDI data being pushed and pulled appears in their correct places in all of the documents.



Translation: Translation software convert those documents and their data to and from the EDI standard.

Network: A secure reliable network is needed to be able to exchange EDI transactions with trading partners.

Support and Maintenance: requirement to implement an EDI system and keep it running smoothly.

In a traditional EDI solution each of these components can be sourced through various providers that specialize in providing these services or software products. Or can be done completely, within the Information Technology department of the respective organizations.

2.3.3 Disadvantages

EDI documents are not necessarily human-readable. Businesses often find that using EDI messages can add time and complexity to supply chain and partner on-boarding processes. But since they are standard for B2B communication, it would be difficult to stop using EDI. The question facing many enterprises is how to stay innovative and agile while still retaining the ability for B2B communication using EDI message standards. [13]

The following are the main disadvantages of EDI.

Expensive: Even though that EDI offers substantial cost savings, for smaller businesses re-designing and deploying software applications to integrate EDI to existing applications can be quite expensive.

Standards: Many organizations also consider that EDI have too many standards. This limits smaller businesses in trading with larger organizations that uses an updated version of a document standard.

Initial setup is time consuming: Not only is it expensive to deploy an EDI system, but it also takes time to set up the necessary components.

System electronic protection: EDI also requires a heavy investment in computer networks. It will need protection from viruses, hacking, malware and other cyber security threats.

Proper backup: EDI needs constant maintenance since the business depends on it. Robust data backups must be in place in the event of a system crash.

2.4 APIs Changing the B2B Landscape

The demand coming from the rapid changes in the technologies and integration methods, B2B integration methods also got the pressure towards changes. With the advent of big data and mobile computing, there arised a need for real-time integrations to get the visibility in Supply chain business processes and dat. This need for real-time connectivity is will lead the developments of future B2B integrations.

These changes create an evolutionary pressure on supply chain partners to grow their IT systems at a faster pace. The paper [5] predicted big operational shifts over the next five years. A shift from point-to-point integration between partners to real workflow, with an emphasis on reusable elements, along with commitment to high availability and integration beyond transactional data are the operational future. More rapid response needs, enabled by newer messaging technologies (web services), improved business visibility and collaboration in troubleshooting support issues define this shift [5].

One of the main considerations in collaboratively sharing the information across the supply chain partners is the security and reliability. The traditional enterprise systems which were developed in-house or installed in secure in house server environments definitely meet these criteria. In addition to that, secure file transfer protocols (SFTP) are used when transferring the information across the partners. Though new technologically advanced services like email on cloud (as provided by Google and others), many organizations, even the technology organizations are still favorable to use the traditional installed Outlook methods. But this evolution is happening in parallel. With some early adopters, the options available in Cloud computing is creating an explosion in digital connecting world.

When compared to the earlier two decades, where some of the B2B partners can still afford to wait to see the maturity in a specific technology adoption, with the current

wave of technologies like Big Data, IoT, API, businesses may no longer afford to wait. It is mainly because, earlier technologies are seen as business enablers. But these technologies are making it possible to create new business models. That's why there is a need to quickly adapt to these methods. Even a medium scale enterprise contains thousands of touch points amongst the partners. These touch points not only compromise the efficiency, but also become show stoppers in the current agile world. Considering these ground level realities, and business aspirations into consideration, there is a need to aware and actively articulate the advantages API based solutions bring. It should elevate itself from Developer-to-Developer communication to Business-to-Business dialogue across the board room members. APIs have the potential to increase agility by de-coupling and exposing business processes. This leads to improved machine to machine communication, which increases the efficient and transparent communication amongst channel partners.

Digitalization is always changing. Digital supply chains continue to evolve, with enterprises realizing the necessity to interact and provide services to partners and customers via digital channels to maintain and strengthen competitive positioning in an increasingly complex digital business ecosystem. Given the rate at which digitalization is driving changes in business processes and customer engagement models, IT just cannot afford to follow traditional approaches for resolving new B2B integration issues. Change is inevitable and IT needs to adapt and effectively respond to B2B integration challenges created by the rapid rise of digitalization. Legacy electronic data interchange (EDI) solutions offer limited flexibility and are inadequate for meeting increasingly critical business requirements, such as rapid trading partner onboarding, end-to-end visibility and monitoring. APIs are looked upon as an extension for B2B integration capabilities.

Application programming interfaces (APIs) has emerged from a development technique to a business model driver and boardroom consideration. An organization's core assets can be reused, and information can be shared, and monetized through APIs that can extend the reach of existing services or provide new revenue streams. While EDI remains the most robust approach to B2B integration, APIs are gaining ground, because of the simplicity and flexibility of implementation and mobile-friendly nature. An "EDI+API" combination can be used to extend B2B processes to mobile channels, thereby allowing mobile applications to participate in and support specific sub-processes,



such as placing, receiving, and acknowledging orders via mobile devices and access to and monitoring of data transfer-related information.

There are several characteristics which enable the positioning of APIs as an appropriate means to B2B integration involving digital channels. The lightweight and developer-friendly representational state transfer (REST) APIs can support real-time (synchronous) B2B information exchange across a range of applications, devices and networks. With digital channels increasingly becoming an integral part of multi-enterprise process automation, enterprises need a more agile approach enabling real-time data transfer, as well as simplifying onboarding and implementation processes. Enterprises can exploit APIs for developing capabilities that would be otherwise difficult to implement with an “EDI-only” approach.

API way of integration for information exchange between organizations can be seen replacing some part of EDI in B2B world. But still, the other forms of collaboration like e-mail etc., and also the other integration option EDI would exist along the way.

With the advent and popularity of APIs, it is speculated that EDI shall be left behind sooner or later. There is a good possibility for companies adopting APIs business to business integration will become easier and message tracking efficient owing to its analytics capabilities but it’s too early to say EDI will be a thing of the past. At this stage, APIs can only replace a subset of EDI transactions and it will probably take about another decade to see the retirement of EDI that too when all organizations unanimously decide to change the business processes at their end to adopt APIs or any similar technology that will crop up in future as the mode of B2B communication.

Consider the example case of how API can be used between a manufacturer/Customer and Supplier. Customer sends message to the Supplier to order certain items. Supplier accepts the order and sends advance shipment notice message to the customer indicating that the shipment is on the way. Supplier then sends invoice to the customer as show in Fig 3 below.



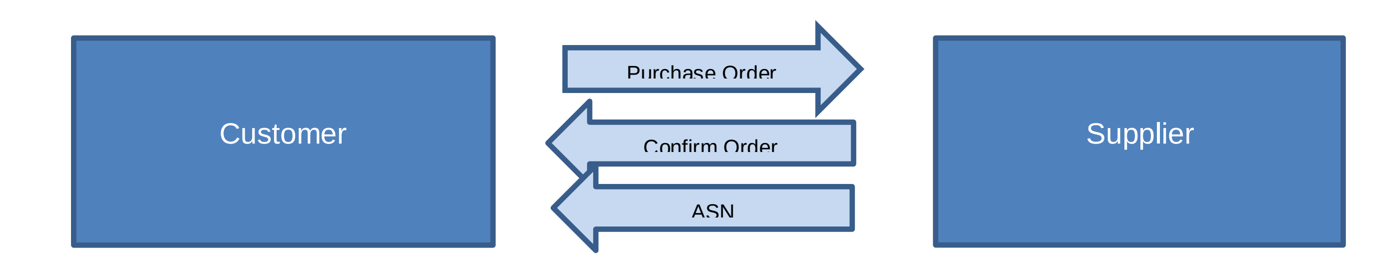


Fig 3 – A typical message flow for Purchase Order document

In this case, customer can develop a purchase order API so that supplier can directly call it and post its order. The intention will be to have the same purchase order API used by all suppliers. But, the practical challenge here is that not all customers send the data in the same way. There is always some variation in the purchase order data from customer to customer which gives rise to different data mapping in EDI.

In this scenario, a meaningful implementation of API will be to develop and expose order status API so that supplier can just call the API to get the status of the order real time. Looking at the developments happening all around, it's quite natural for an organization to jump on the API bandwagon. But the right strategy for API implementation by an organization could be to introspect its various B2B processes and identify the areas where it truly makes sense to go the API way to reap its benefits of faster response and many others. As stated in the example above, APIs can be created for getting the real time status of a shipment which shipper can access at any point of time. Similarly, the customer can check the status of its order by just calling the order status API created and exposed by the manufacturing company as shown in Fig 4. So, EDI and API can complement each other and they together can make b2b communications all the more efficient.

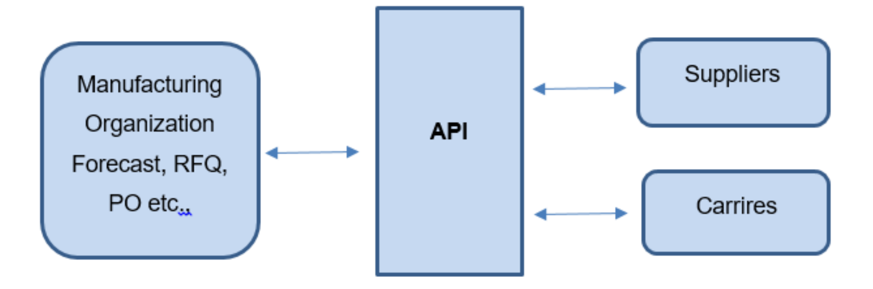


Fig 4 – API as an interface



2.5 Analysis of Pros and Cons of APIs

The most crucial advantage of API in B2B Networks is interoperability. Interoperability allows various IT software and system applications to use and swap data across a service provider, facility or client, irrespective of the vendor or application. Therefore, APIs are an excellent complement to B2B interactions and can be leveraged along with existing B2B technologies.

APIs connects an organization with its suppliers and partners giving the real-time data flow experience and visibility. The performance and the user experience can be easily improved with APIs. The publisher or the provider of the API can make the data available for the supplier/customers/partners. This data will be consumed by them and can be integrated directly to their ERP and can communicate back with APIs. This facilitates the real-time vision in the processes of supply chain.

Bridging and agile management of data sources is another plus point with APIs. In current systems and software, data points are connected with each other via APIs (through single sources) allowing the necessary data to be available to the relevant stakeholders as and when it's required. Organizing and managing these data sources becomes agile when APIs are integrated as they can exchange information with the processes and management systems of the inherited data. API works like a retailer providing a customer with the right information at the point of sale [6].

APIs reduce security problems in the development processes, as the implementation of an API will mean your company's developers will have to compartmentalize the business applications and processes and reinforce the most sensitive internal systems.
Being said about some of the advantages of APIs, following is some comparison between EDI and API.

EDI	API
Partner oriented	Application or User oriented
Based on Industry standards	Based on technical standards
Business application friendly	Friendly with mobile devices too
Deployment consumes some time	Fast deployment



Standard message formats (POs, Invoice, ASN etc.,) Driven mainly by standard bodies.	Ad Hoc message formats. EDI formats can be used in principle but only applicable for basic implementations. Driven mainly by service implementer.
System of record	System of engagement
Partner on-boarding requires technical and business workflow	Partner on-boarding is typically simpler
Services are well defined and do not evolve regularly	Services are defined via APIs which require full life cycle management
Business agreements are often required	Usage conditions are defined unilaterally by APIs
SLAs are commonly in place	SLAs are not the top issue

3 Explaining APIs

This chapter touches how the APIs have evolved from the early days of software management using Remote Procedure Calls to the webAPIs. Different technological choices along with encapsulation mechanisms are addressed in detail. This chapter also touches the different styles of webAPIs and deployment mechanisms. Where relevant, appropriate commercial vendors are referenced.

3.1 Definition of an API

Though many forms of definitions are available for an API, in a simple terms, an API, stand for “Application Programming Interface”, is a tool used to share content and data between software applications.

Though the usage of APIs has become extremely popular amongst the interconnected world of 21st century, these concepts date back even much before the days of personal computing. According to Martin Bartlett - The basic principle of well documented set of publicly addressable interaction points that allow once system to interact with another system has been part of the software development since the earliest days of utility data processing [7]. However, with the advent of distributed systems, interconnected systems, the importance and utility of these basic concepts has increased dramatically

3.2 History and Evolution of Web APIs

Similar to evolution of many concepts and objects, in computing world, APIs evolved and emerged into the current form of web APIs

3.2.1 Remote Procedure Call (RPCs)

When two software pieces are developed using the same programming language or same platform, the advantage of re-using the software elements is obvious. However, when the applications are developed using two different languages or platforms, reusing the existing software posed a unique challenge to the development community.

As shown in Fig 5, this is where the IDL (Interface Description Languages) came into usage. These are specification languages that describe the software component's Application Programming Interface. Based on these definitions, "Stubs" are generated for client programs, and "Skeletons" are generated for Server programs. Communication between Client and Server programs are passed through Client - Stubs – RPC Run time – Skeleton – Server. Packet based communication is used between the RPC runtimes.

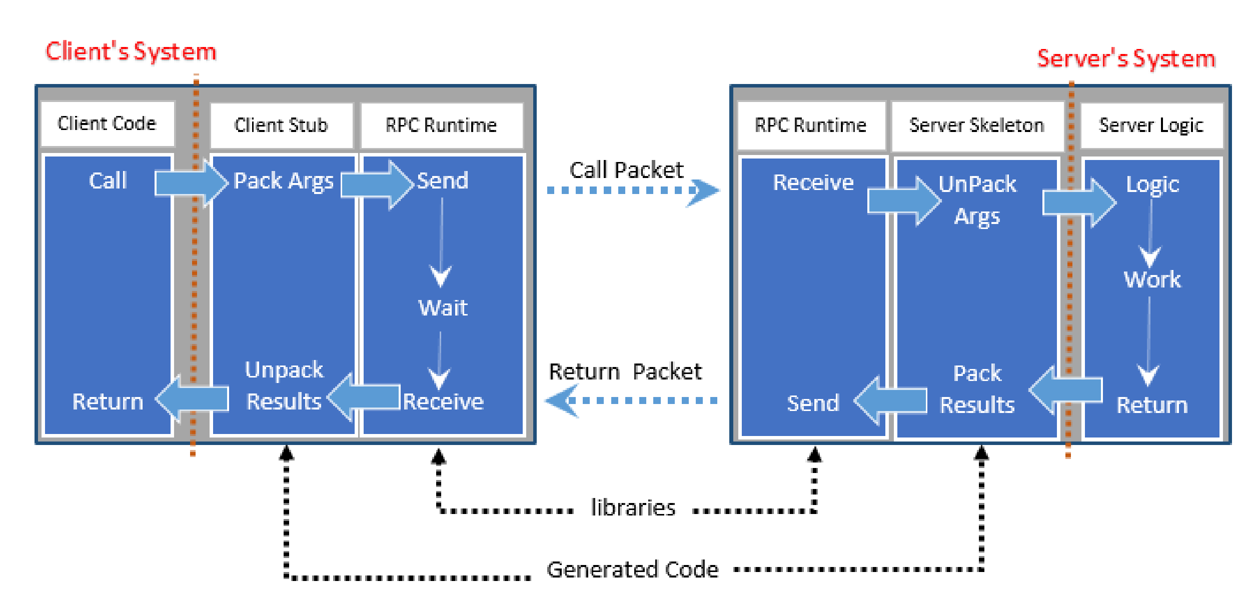


Fig 5 – Invoke remote procedure using RPC

Some of the commercial vendors in this era include Sun's ONC RPC, The Open Group's Distributed Computing Environment, IBM's System Object Model etc.,.



3.2.2 Common Object Request Broker Architecture (CORBA)

With the advent of Object Oriented Programming, during 1991, Object Management Group (OMG) has come up with mechanism to normalize the method calling semantics between the application objects. These application objects can reside in the same address space (application) or in the remote address space (same host, remote host on a network).

In this architecture, as shown in Fig 6, the RPC runtimes on both client and server side are replaced with ORB (Object Request Brokers) that are generated based on the language or platform on which the client and server applications are developed. The communication protocol between both the address spaces has been IIOP (Internet Inter ORB protocol).

Effectively, in this architecture, ORB act as intermediary between both the systems and IIOP is used as communication protocol.

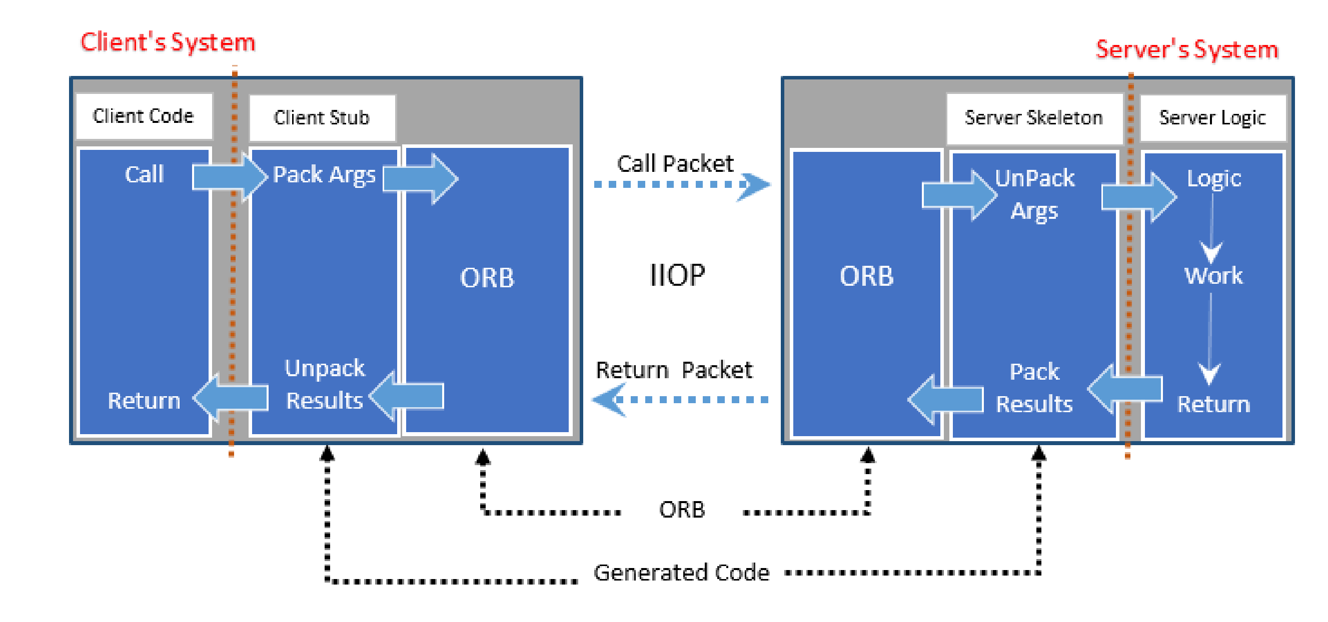


Fig 6 – Invoke remote procedure using CORBA



3.2.3 Java RMI (Remote Method Invocation)

When the Java based environments facilitated the direct transfer of serialized objects (or classes in Java world), and distributed garbage collection, Java based Remote Method Invocation came into picture. During the initial days of Java development, this invocation was limited between two Java Virtual Machines. The protocol for such implementations is JRMP (Java Remote Method Protocol). In order to support the code running in non-JVM context, programmers later came up with CORBA version.

Across the developer community when the word RMI is used, it is synonymous using the JRMP, whereas when the word RMI-IIOP (read as RMI over IIOP) is analogues to implementing the CORBA implementations. Communication in this architecture is depictured in Fig 7.

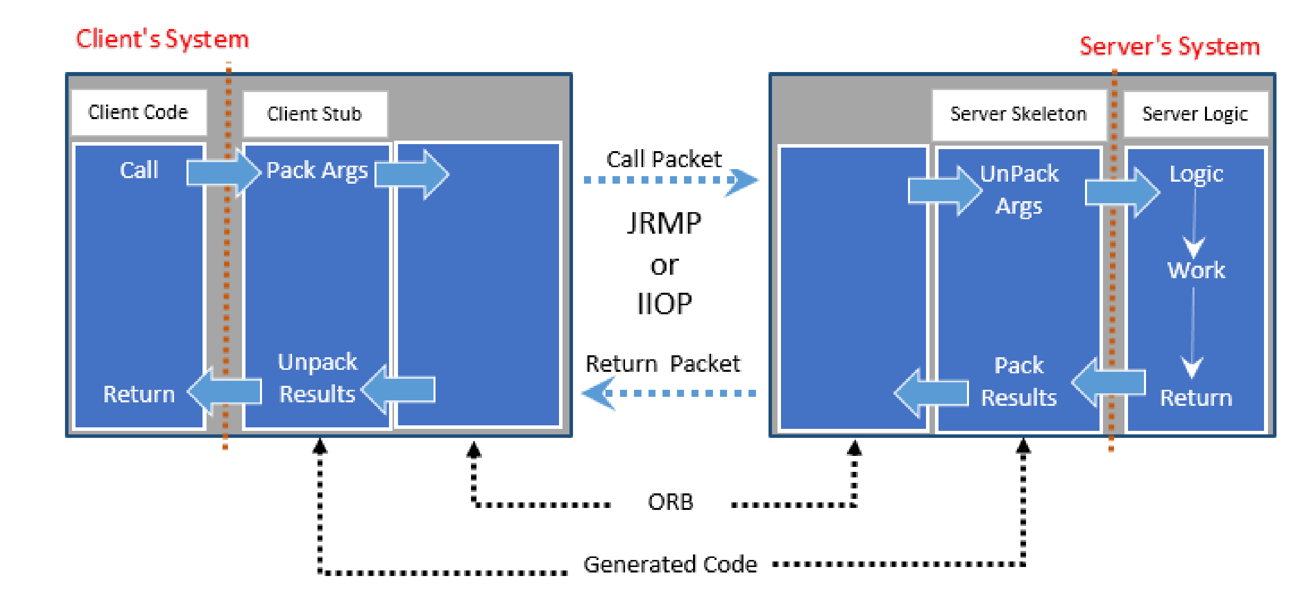


Fig 7 – Invoke remote procedure using Java RMI

3.2.4 EJBs RMI

With further evolution of Java technologies, logic of the remote application are completely encapsulated in Enterprise Java Beans. As shown in Fig 8, An EJB objects which is generated acts as a wrapper and makes it completely unaware of the communication protocol. This approach has led to advent of application servers which could communicate the logic embedded in multiple applications that are spread across the enterprise



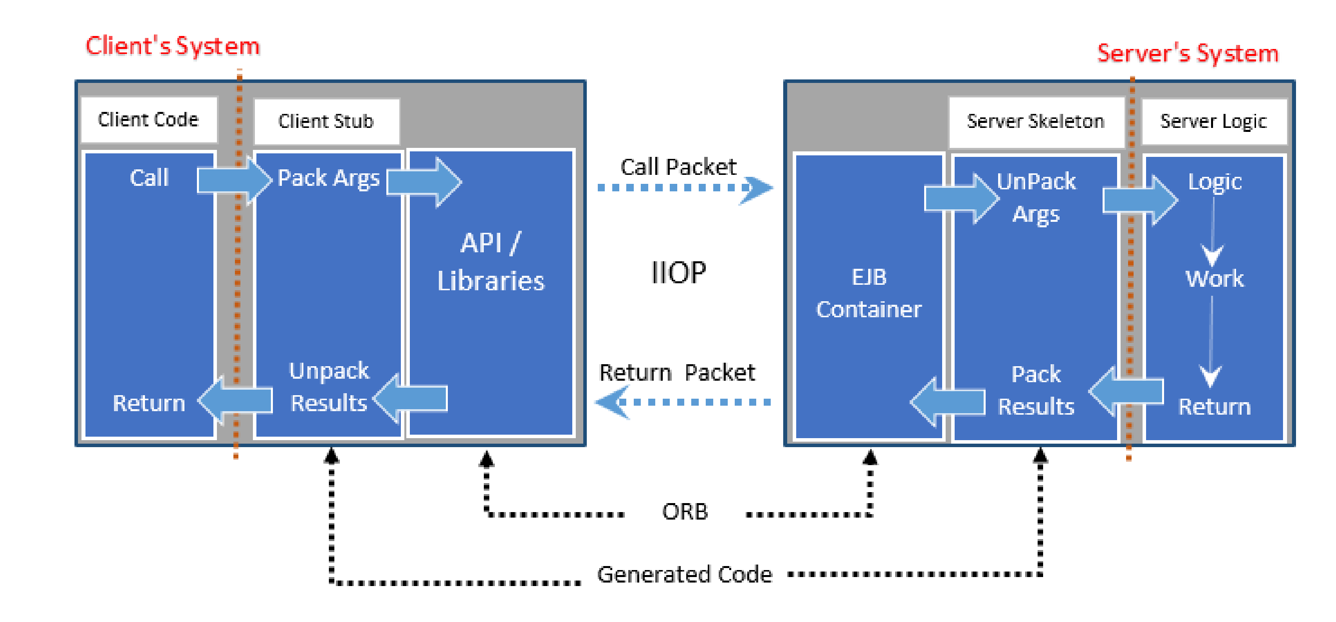


Fig 8 – Invoke remote procedure using EJB

3.2.5 Web Services SOAP

When XML is started to be used for representing data, the approach has become completely platform agnostic. Communication has become has programming language independent and platform neutral.

As depicted in Fig 9, using the protocols like Simple Object Access Protocol, arguments for remote method invocations are passed as XML message. On the server side messages are unpacked by parsing the XML message and response is constructed again in the form of XML message and sent back to client.



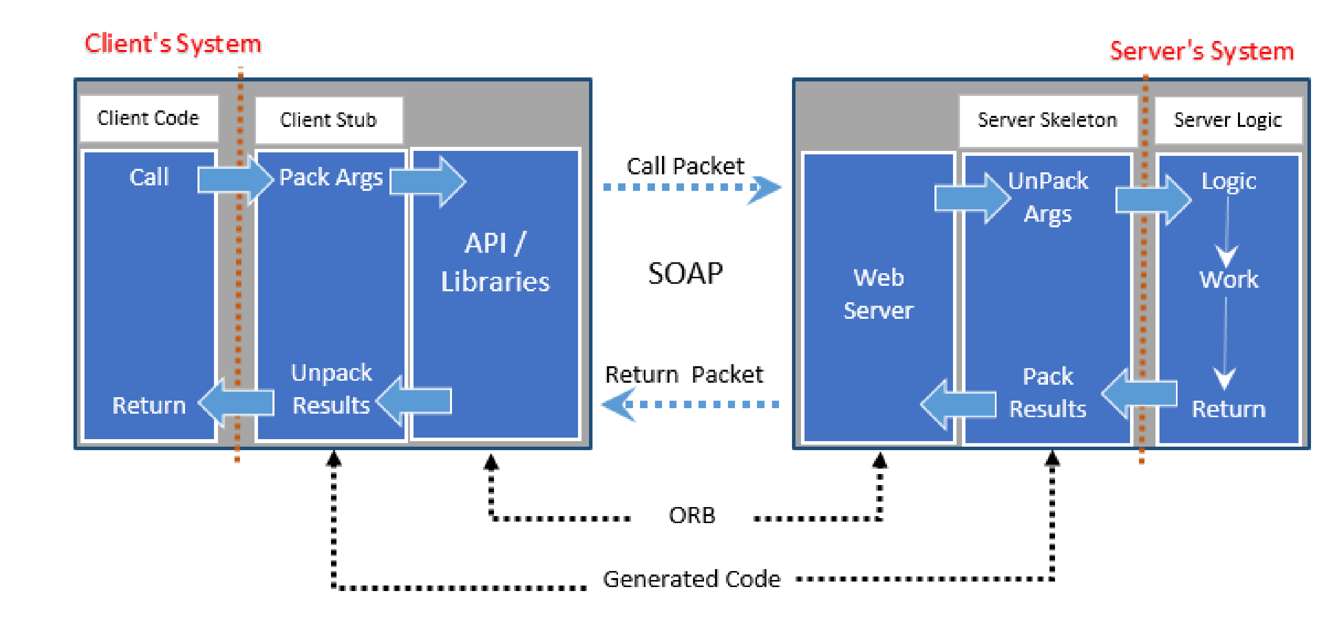


Fig 9 – Invoke remote procedure using SOAP

Availability of these technologies also resulted in creation of Enterprise Service Bus (ESB) where different underlying applications are exposed in the form of services. Business Users seen these approaches as "Technology Agnostic" and started focusing orchestration of business processes by picking the touch points as services.
Microsoft BizTalk servers and new players like Cordys (now acquired by Open Text) started offering the platforms that challenged the giants like IBM and offered possibilities to build workflow based applications with very short go-to-market times.

3.2.6 Web APIs

By this time in the history, web applications become ubiquitous. A web URL become synonymous the street address for every business. But the initial days of website content is human consumption.

In the early phases, as represented in Fig 10, most of the user request processing is done on servers side is pages are rendered. In other words, all the logic is executed on server side, and results are put in a pre-defined template and presented back in browser using HTML.

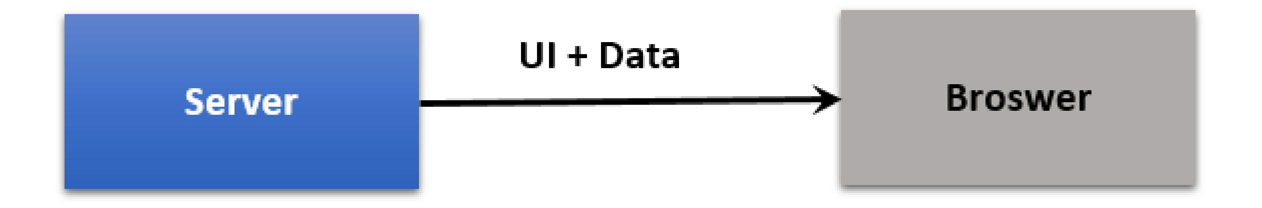


Fig 10 – Simple Web Application

But in the next phases, as shown in Fig 11, with technologies like AJAX and Dynamic HTML, the web page that was already loaded into the browser started talking to sever and update itself. Though the technology become complex when compared to earlier versions, development community took advantage of these, and started building the main page of the UI only once and it update itself by talking to the server and processing of the business logic is moved to the browser.

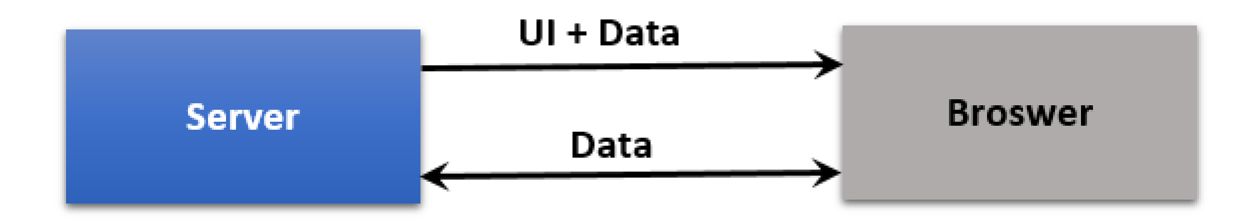


Fig 11 – Dynamic Web

This experience is exactly same like installed an application in the mobile The only difference is, HTTP protocol is used to serve the UI in the web applications as shown in Fig 12.

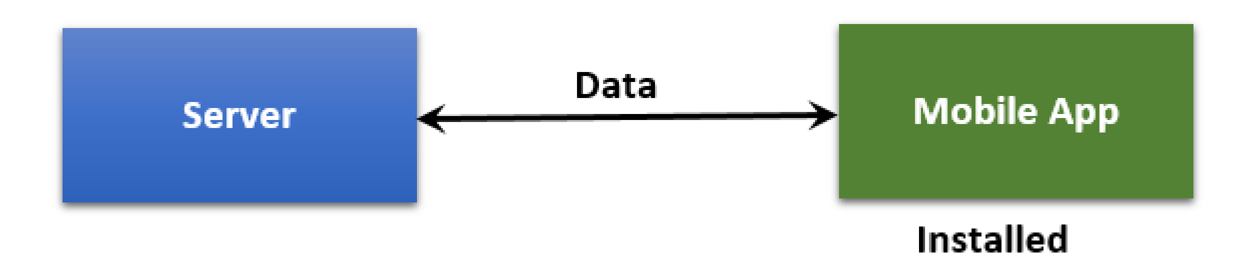


Fig 12 – Mobile Applications

When the next generation of protocols like REST are available, even the backend services started talking to each other in a very user friendly format over HTTP. This paradigm made the whole computing even more interesting. Applications could talk to various services which may be running over the cloud using the "Application Programming Interfaces (APIs)". There is no limitation on the number of APIs a server is talking to as depicted in Fig 13. Rather the interaction became so dynamic and user interaction or logic based.

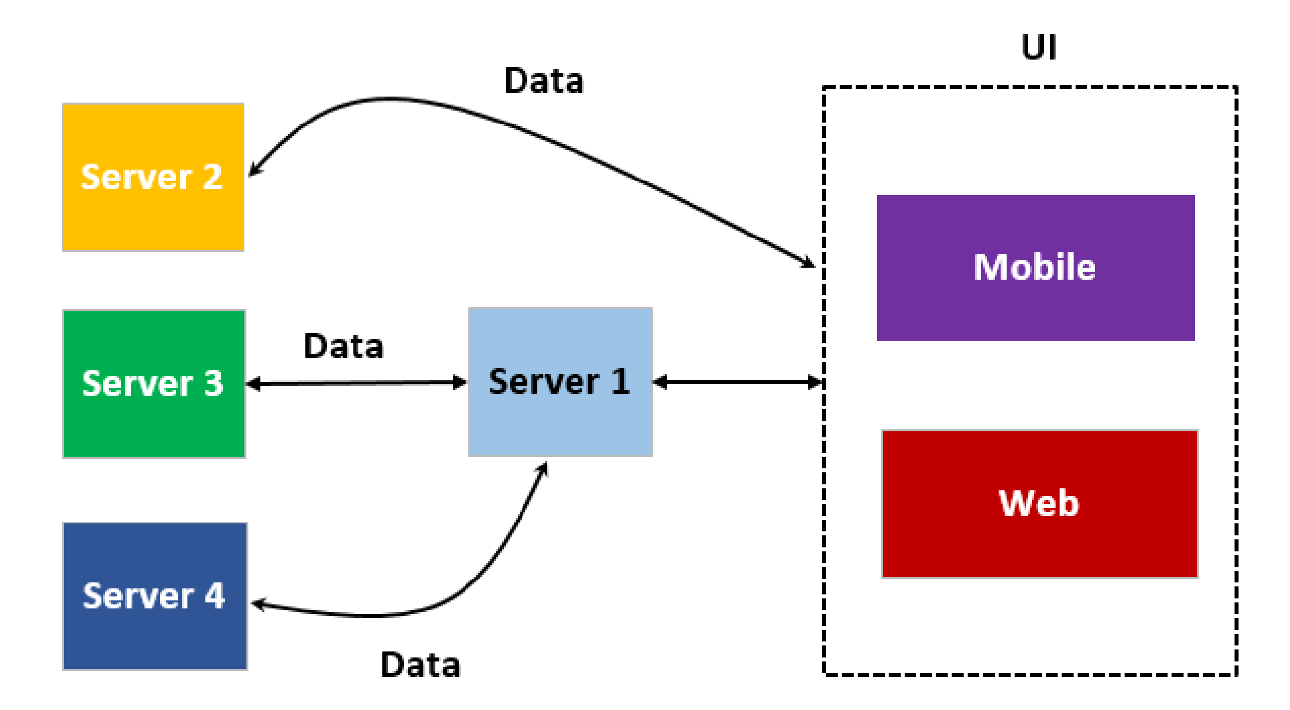


Fig 13 – web APIs

The real power APIs is their "Simplicity" for the machines and programs to talk to each other regardless of their architecture, location or even the programming language.

Business community who ones built myraid of technologies to leverage the already invested technologies cant starting think of "Dream Come True" for integrating teh divesisified sources.



3.3 Elements of API Value Chain

In a typical goods or services industry, the value chain from production to consumption can be depicted as in Fig 14. This represents the traditional industries value chain.

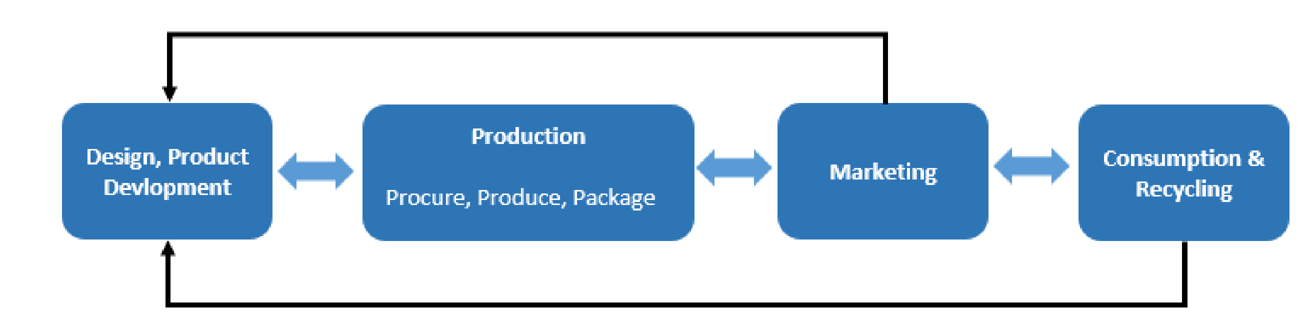


Fig 14 – Value Chain in Manufacturing Industries

However with relatively maturing Software Engineering practices, and also when the business models of businesses are getting redefined to open new business channels, the API value chain is still emerging.

In his work, Kin Lane, API Evangelist noted that, “An API industry has risen up to support businesses to develop, implement and succeed in their API journey. This industry is creating value by supplying goods and services needed for success. These suppliers make up an API industry supply value chain”.

With very low entries of barrier, and being a knowledge intensive industry, both well-established giants and startups co-exist in delivering innovation solutions to both industrial as well as niche customers. An example of this represented in Fig 15.

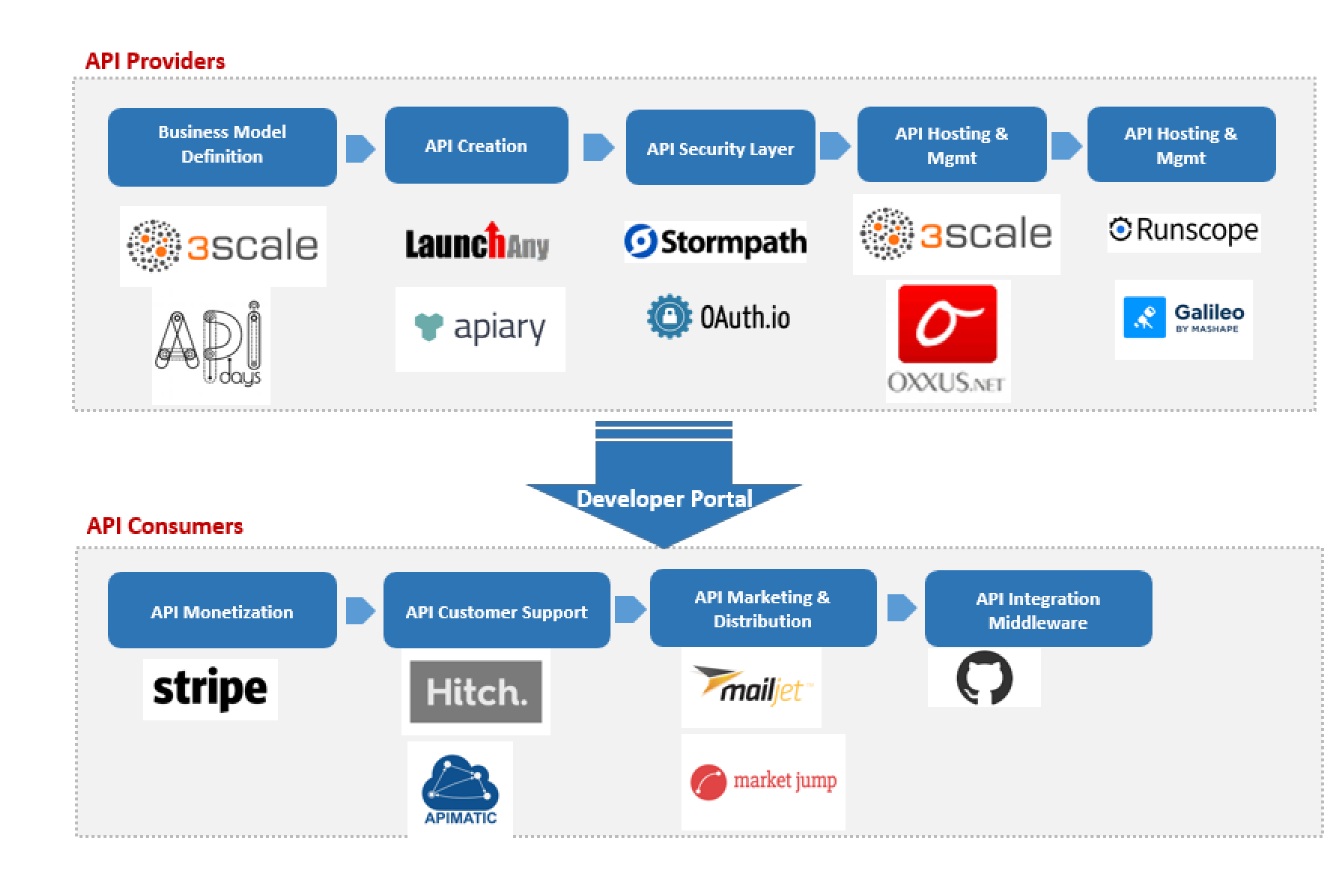


Fig 15 – Value Chain in API Industries

By evolving into such a stack of services, APIs have been elevated from a development technique to a Business Model driver and for boardroom consideration. An organization's core assets can be reused, shared and monetized through APIs that can extend the reach of existing services or provide new revenue streams.

3.4 Business Assets

In traditional brick and mortar world, real estate facilities and machinery are considered as assets and accounting methodologies are well established to associate a monetary value for them. As the service industries emerge, key personnel working in the organization and intangible services they provide are considered as assets. In the digital economy, APIs have emerged as the business assets. Similar to the way the express highways facilitated the quick movement of goods and contributed to the economy, in digital world, APIs are acting as super highways in establishing the connectivity and make the business community focus on process possibilities. By leveraging the already existing computing assets, these new class of assets started forming as pillars.



3.5 API Providers – App Developers – Client Apps

Looking purely from functional perspective, as shown in Fig 16, API value chain can also be looked in as “Backend Systems – API Providers – API Developers – Client Apps – End Users”.

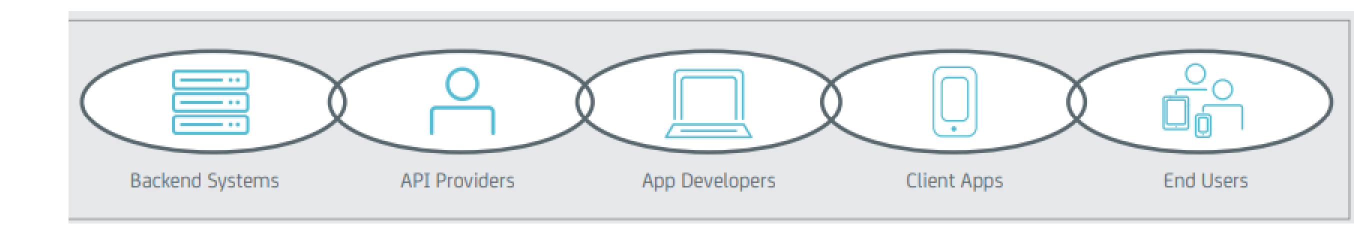


Fig 16 – Structure of API industry value chain

To build new APIs and bring value to customers, there are many API prototyping tools. Most popular tools available in the market are: Apiary, Raml and swagger.

Great value has been generated for many organizations by adapting to this strategy can be put into areas like, generate new revenue directly, extend customer reach and value, support sales and marketing activities, simulate business and technical innovation.

API Styles: The kind of interface published by the API is affected by many considerations. Considering the different purposes for which APIs are used, different styles of APIs can be classified as**,** Web Service (similar to tunneling)**,** pragmatic REST (or called as URI)**,** hypermedia (or true REST) and, event driven (falls under the bucket of IoT).

3.6 API Strategies

Looking from the way, the business partners can interact with each other and gain the insights, different strategies can be classified as public, partner and private.

Public - Most commonly these are also referred as open APIs. These allow companies to publicly expose information and functionalities of one or various systems and applications to third parties that do not necessarily have a business relationship with them.



Main advantages from this kind of APIs are: Delegated R&D, Increased reach, traffic and create new revenue stream

Partner - Partner APIs are used to facilitate communication and integration between a company and business partners. Main advantages from this kind of APIs are Value added services, up sell and a must have for business partners

Private - Private APIs are used internally to facilitate the integration of different applications and systems used by the organizations. Main advantages from this kind of APIs are rationalized infrastructure, reduced costs, increased flexibility – real-time business and Improved internal operations

3.7 API Standards

Seamless integration amongst the supply chain partners has long been a utopia for many enterprises. Organizations would like to leverage this integration and gain the Supply Chain Visibility through a centralized dashboard. Though there are standards and technologies are available for B2B message exchange since more than two decades, efforts to link the IT systems and procurement processes have been tardy, complex and disappointing for boardroom that expects a quick turnaround. The ability to on-board and conduct transactions with suppliers is hampered by a lack of integration and reliance on error-prone, tedious, manual processes with resulting excessive labor costs. Ultimately, the business itself will suffer.

However, with API way of integration, businesses now have an opportunity to participate in communities of buyers, sellers and partners who can rapidly interact with each other in a collaborative and transparent manner.

Over the evolution, there came myriad of methods for business partners to connect electronically. One is Electronic Data Interchange (EDI) and its various standards. Other methods of B2B integration include RosettaNet, GS1, EDIFACT, cXML, ebXML, extranets, Trading Hubs and Value Added Networks. Besides these standard approaches, there are countless home-grown solutions developed by in house by the big enterprises.

None of these technology choices have been efficient enough to provide real time connectivity with the trading partners. They are time consuming and also expensive. This is

mainly because of disparate enterprise systems, business processes, business rules on both supplier and vendor side. As a result of this, each connection with the partner is to be configured considering the needs. In turn, these connections require investment in the additional hardware, software, customizations, integration work which increase the total cost of ownership. Hence, most of the trading networks limited to partners who can invest the time, and able to maintain the connectivity.

With the rise of internet, there is hoped for an open, public network that would provide a way where the EDI could not reach earlier. However, same challenges seen in EDI are still present - Before establishing the connection, it need to be negotiated, configured to comply with the partners' need. To initiate the process, procurement and IT managers need work with their counterparts, and agree on the communication protocols. After that, they need to keep track of the data and status of each message. Some organizations opened up Collaboration Platforms using which partner organizations can log in and keep track of the items. Information from collaboration platforms is to be further exported back into the data formats required by the partner systems and develop the import / integration mechanisms.

With the increased globalization, there a constant pressure on all types of businesses to offer product at a competitive price to the customers. That forced the organizations to source the materials / components / or design activities from multiple parts of the globe. At any time there are thousands of suppliers and buyers who are getting engaged in the process. These organizations are producing multitude of products, with a very unique specifications as well as maintaining the complete tracking number from the source – sub-contractors – manufacturing unit – end customers. This adds up to an extreme amount of complexity.

With the innovation of API based interfaces, one connection is offering the endless possibilities to integrate with business partners. It moves beyond from the simple movement of data to deeper process integration. Partners can scale up their operations with real time integration without acquiring any specialty software, hardware or expertise. This technologies really make the supply chain partners as a logical extension of enterprise.

4 How to Add APIs to an Organization's Digital Tool Kit

In the digital economy, neither individuals nor organizations are immune from not using an API. Either it explicitly used as part of a process or some of the underlying elements make an API call, these are widely prevalent. On enterprise scale, when any of the SaaS offering are used or any infrastructure services hosted in cloud are used, APIs are inherently used. As a consumer it is always and rewarding experience. On the flip side, when the organizations plan to provide the APIs, their intention is to leverage the existing assets in the form of data, application logic or transactions information to their partners, suppliers and customers depending on the usage. This part of the thesis focuses on an organization preparing itself to move towards opening up the supply chain APIs to their suppliers.

As a API providers, there are few governing principles one need to be considerate about. The rudimentary element of this is the API readiness. With multitude of business partners and uncontrolled web, when governance aspect is not well implemented, it can easily spin out of control. Having an well-established governance plan that explicitly articulates the standards using which connectivity is established, usage policies using the security policies, and the governance elements that cover the versioning the successive version of APIs is a very important element. This also mean, touching the external aspects that are beyond the description of API, SLA with which it is offered, how the API is going to registered in an external directory that is searchable by the business partners and how the data level security elements are implemented, and how the usage metrics are collected. These usage statistics play a key role in deciding which of them gaining traction and any load taking mechanism that need to be kept in place.

In addition to building the APIs, they also need to be published on a very robust platform. Platforms typically govern the elements of ensuring that APIs are offered on high available foundation, that is reliable and built to scale up. These platform should offer good flexibility to connect to any of the available database platforms, applications available in the organization. The platform should also be flexible to call other APIs which are available within the organization, public domain or in a controlled domain. All of this leads to the API-led connectivity concept, which will be discussed in further depth below.
Considering the study done in an anonymous Manufacturing organization in the Supply chain management area, in this chapter are presented the process and structure fol-

lowed. In order to get API to the supplier collaboration tool kit, the strategy of the organization should be clear about the future and how to move ahead with the current infrastructure which support the existing integrations and collaborations. Once the API mindset is made up, it is required to understand the requirements and costs for the API development, deployment and maintenance. This needs to be discussed well ahead within the supply chain organization along with the IT teams. These requirements are further discussed as we are moving ahead in this chapter. After these investigations are made, it is also important to understand the API awareness in the Supplier and partner ecosystem. Being rather a new integration option which is evolving and emerging, it is also important to understand and make the business case for the investment. This has been conducted in this anonymous organization and the results are discussed in detail in the next chapter.

In this chapter presented are the building blocks which need to be put into place to develop, deploy and also maintain APIs successfully. These are represented in Fig 17.

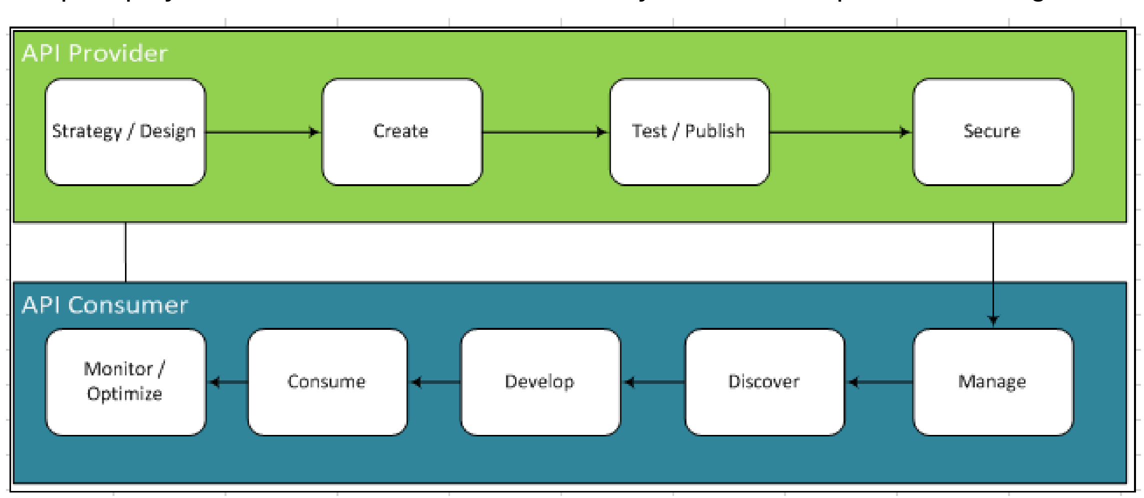


Fig 17 – API Life Cycle Management

4.1 Identifying and Defining APIs for Supply Chain

Using the current integration technologies, it takes time and cost in order to add a new supplier to the supply chain. The code needs to be tailored for each supplier integration and, depending on their system suppliers also needs to do the same. This becomes a mini project which has typically some integration specialists, developers and business analysts form both the organization and supplier. This takes typically from weeks to



months depending on the complexity of the business processes and standards to complete the end to end testing and fixes that the data flow is clear. Supply chain visibility in this case depends mainly on the ability to collect the telemetry data from the supplier systems which might not be reliable and gets complicated of there are several integration platform vendors are in between. Because of this, organization will not get the real-time end to end visibility of supply chain. Also, in this, the maintenance and integration costs are very high and it adds up with more number of suppliers with the increasing data flow.

API could be a good answer to handle the lacking of the traditional integration systems. But, it is very important to have a view towards future and analyze carefully on which functionalities can be opened for APIs before jumping into conclusion of including everything. First it needs to be understood on what are the messages which can be moved to API collaboration and why. To start with in the supply chain management, messages like forecast collaboration, request for Quotation, purchase order, order confirmation, traceability codes, invoice, pick up requests and ASNs could all be considered for this transformation. Once these are clearly identified, it is easier to move ahead further.

4.2 Participants

There are number of participants who needs to be included in the process of defining and development of APIs. Starting with the business stakeholders who deal day-to-day with those specific transactions, solution managers who are already aware of the architecture and message structures, IT teams who will publish and maintain the API management, business owners who will maintain the governance models etc.,The team should be identified before moving towards designing the API specifications. Participation of the following teams must be considered in this process.

4.2.1 Keep in Mind about API Consumer

API consumers in this context could be supplier side integration Developers who will consume the APIs being exposed. Ultimately the API Consumer facing documentation is best placed in a Developer Portal where developers can sign up for your APIs and test out your API. An approach where the consumer facing definition of the API could be defined first using a simple modelling language such as Swagger gets you off to a good start. A Swagger definition can then be used to build your API Consumer facing documentation e.g. using SmartDocs. One of the key benefits of starting with Swagger is that



its machine readable and so can be re used to automate the generation of other documentation.

A good API documentation should be an easy to understand even for those who does not have prior knowledge on what that API is meant for. There should be some high level explanation of the intent of the API mentioned clearly. Also, the usage of the API should be specified and preferably with some examples. The documentation should always contain information about how to start with this API as not all the developers might be familiar with it. It would be good to share materials and information about the check list on what is required to start with the API and be able to consume it [11]. This description and documentation of API is very important and self-clear in such a way that the consumers of the API needs not be trained or communicated about this separately.

The following tips can be applied to ensure the information is presented in the most favorable way possible: Document each API call separately, with parameters and their values explained. Provide examples of each call being made accompanied with details about the requests and responses. Make it easy to find what is needed by adopting a menu of links. Explain and exemplify request headers, API responses and error codes just as GitHub does. It is very helpful for newcomers trying to understand how an API call's behavior can change based on certain HTTP request headers. Provide developers with an ability to make live calls right from the documentation page Create clear tutorials for such complex issues as authentication. These will be much looked up by visitors having no previous experience with the resource you work with every day Illustrate the best of your API by providing code samples in the most popular programming languages. [8]

Also, API management platform information needs to be clarified with the suppliers beforehand so that they are aware about the authentication and authorization mechanisms. When on boarding new suppliers for integration this basic package of the API know how needs to be distributed well ahead.

4.2.2 Product Owner

The owner of the business requirements behind the APIs might not care about the low level technical details of the API, but will want a higher level overview of their overall API Catalogue. API catalogue includes all the APIs belonging to a business purpose. Product owner should be also considering of developing and building a governance

model in place which can be further agreed and communicated internally and externally. They will also want to be able to see a consistent approach to how resources are being exposed across APIs. Product owner should also consider about the API management system well beforehand.

4.2.3 API Maintenance Team

How to manage the APIs published should be a thought which needs to be put well ahead in place. If the organization is having their own IT department, then it is good idea to set up a dedicated and well-resourced API team which would be taking care of designing, launching and managing of APIs. Publishing API is just a start. The team needs to ensure the availability, monitor manage the versioning etc.,

4.3 Think about API Management

API management platforms allow organizations to accelerate innovation, provide shared services and data, and adapt to market and supplier needs. APIs have become the foundation of the fast-moving digital economy. API management maintains the entire lifecycle of the API and tools for all stakeholders. API management is the process of designing, publishing, documenting and analyzing APIs in a secure environment as depicted in Fig 18. Through an API management solution, an organization can guarantee that both the public and internal APIs they create are consumable and secure. [12]

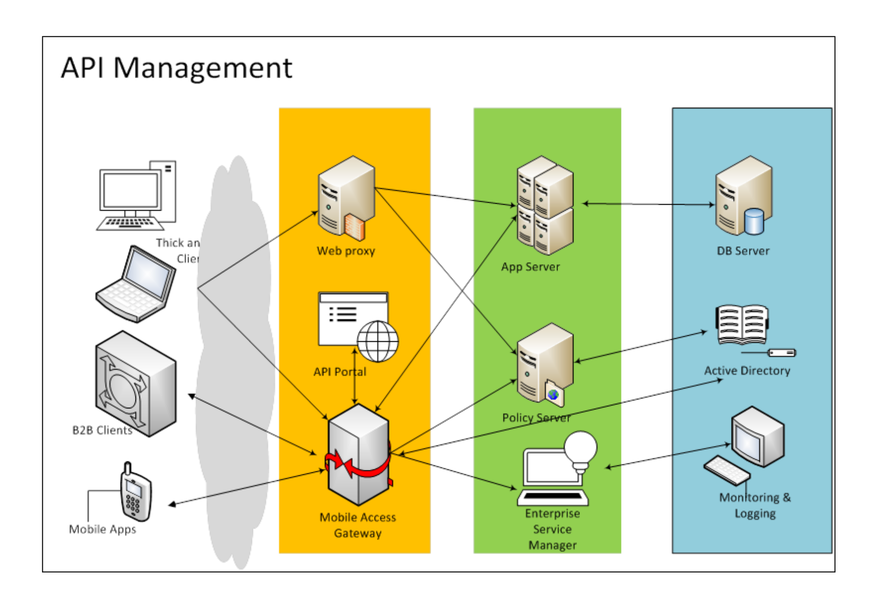


Fig 18 – API Management scenario



API management solutions in the market can offer a variety of features; however, the majority of API management solutions allow users to perform the following tasks:
API design - API management solutions provide users – from developers to partners – the ability to design, publish and deploy APIs as well as record documentation, security policies, descriptions, usage limits, runtime capabilities and other relevant information.

API gateway - API management solutions also serve as an API gateway, which acts as a gatekeeper for all APIs by enforcing relevant API security policies and requests and also guarantees authorization and security.

API store - API management solutions provide users with the ability to keep their APIs in a store or catalog where they can expose them to internal and/or external stakeholders.

API analytics - API management allow organization to monitor API usage, load, transaction logs, historical data and other metrics that better inform the status as well as the success of the APIs available. [12]

The above features represent some of the 'major' offerings that API management solutions provide. The marketplace for API management was $70 million in 2013 and, according to Gartner; this figure is expected to grow at 33.4% between 2017 and 2022, reaching an estimated 3,436.16 Million by 2022 according to Zion Market Research. [15]

These features ensure Full API lifecycle management offering organizations the ability to have a unified platform that tracks each API and integration from end-to-end. In addition, the API reusability component of the platform leads to improved project speed. Hence, API management is becoming an increasingly important solution for an organization that uses APIs or depends on them to deliver services.

There are several API Management platforms available in market which variety of features like Microsoft Azure, Apigee, Mulesoft Anypoint, Kong etc., All these would provide effective solutions for the API lifecycle maintenance.

4.3.1 API Gateway

API gateway is a core feature of an API management platform. With an API gateway, you can design API specs, provide enterprise-grade security, and manage APIs centrally. An API gateway, as shown in Fig 19, sits between suppliers and the API publishing Organization. API Gateway is the front line of guard in analyzing the request coming from



various consumer channels and route them to the services. It plays a key role in authenticating the user requests, rate limit to prevent any Denial of Service attacks, decrypting the data coming from SSL connections on different areas so that server is not loaded. When an API gateway is not deployed as part of solution, then all client requests are directly routed to the services.

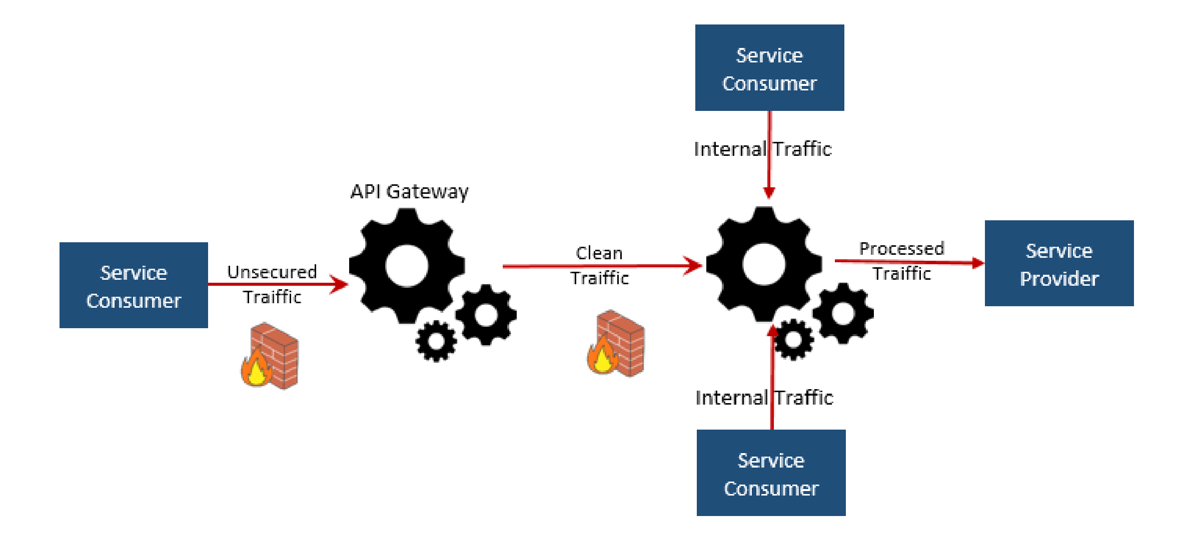


Fig 19 – API Gateway

Though it is technically possible to expose the services directly to clients, it comes with multitude of areas that need to be addressed on the client side. Client applications should keep track of end points and develop the exception handling mechanisms in a robust way so that application does not break-out dump memory. In addition to this, client must know the specific details of how the individual services are decomposed. As a result of this, it would lead to a tight coupling between client and backend. When any of the services are refactored, it directly impacts the client, and it becomes a very hard to maintain the client. In cases where a single operation require calls to multiple services, it would lead to multiple back-and-forth connection hops between client and server which obviously brings the elements of latency and performance issues start popping up. As a general architectural recommendation, every publicly exposed service must be meticulous on areas of authentication, limiting the client rate to address the denial of service attacks, and decrypting services to minimize the load on server. As a general usability, services must expose most widely or easy to implementable protocols like HTTP or websocket. This will limit the choice of protocols using which the requests can be sent, but focus on the content itself. Similarly all the severs that provide these services need to be hardened.



A gateway seamlessly addresses these issues by decoupling the client with these complexities from the services provided by APIs. Apart from these essential functions, Gateways can perform many other functions. Depending on the implementation, those can be chosen for activation. These features can broadly be classified into following design patterns.

Gateway Routing: When a client request need to contact multiple backend services, Gateway helps and acts as a Router. It encapsulates the complexity, acts like single end point service to clients. As a result of this, both the clients and services are completely decoupled and routed through API gateway.

Gateway Aggregation: To perform an operation, if the client application need to make multiple calls to different backend systems, it will increase the need for network resources. And also, when any additional features are added, it calls for additional steps and addition resources. A typical Gateway aggregates the requests. In behavior, it resembles a typical application server that minimizes the number of backend calls that need to be made.

Gateway Offloading: This is one of the very important features. Gateways can offload the some functionalities like certificate management. This approach helps to consolidate similar features or services to one single place. When implemented well, data can be requested on behalf of the original requestor from any HTTP server which are available upstream of the API gateway.

Here are some examples of functionality that could be offloaded to a gateway: SSL termination, Authentication, IP whitelisting, Client rate limiting (throttling), Logging and monitoring, Response caching, Web application firewall, GZIP compression, Servicing static content.

4.3.2 API Security Aspects Needs to be Considered

API Security is usually an afterthought for API Management and Integration platforms. The early pioneers of API management laid focus and emphasis on areas like analytics, monitoring, and lifecycle of their API's. However, it is impossible to create a well though strategy and reliable infrastructure without addressing the API security related challenges. When there is a room for security compromise, it is of little benefit to have the

best monitoring and design solution. Ensuring that infrastructure (and data) is secure is the premise of API Security Management.

API security management is a big part of API Gateway. Secure your APIs using a key, token, and IP filtering. On top of these, establish the policies that address the rate limits, establish the quota limits, and using the techniques like caching take away the risks associated with latency. Simplify and optimize requests and responses with transformation policies. Secure APIs with key, JSON Web Token (JWT) validation, and IP filtering. Usage of rate limits always protect the APIs from overload. Quota limits help in preventing from over usage. Use response caching for improved latency and scale.

Access control is the primary security driver for API Gateway technology, serving as a governor of sorts so an organization can manage who can access an API and establish rules around how data requests are handled. An API Gateway's access control capabilities usually start with authentication mechanisms to determine the actual source of any API calls. Currently, the most popular gateway is OAuth, which acts as an intermediary for accessing web-based resources without exposing a password to the service, with key-based authentication reserved for instances in which the business can afford to lose the data because it's difficult to guarantee complete secrecy of the keys.

4.4 Fundamental Design Principles - Integration Strategy

In designing the solutions and evaluating the solution choices, it is important to keep the solution objectives that need to be achieved. As myriad of systems have been developed and implemented in the enterprise over a period of time, service oriented approach gives the flexibility to reutilize the existing enterprise assets. However, the fundamental principles that need to be continuously evaluated during the course could be:

Simple and Scalable. In designing the partner integrations, the modular the services are made, the simple the integrations will be and higher scalable they will be. With the advanced API gateways available, having module and micro services does not result into an increased network traffic. Rather, they will form a scalable solution with multiple partners.



Address the present, design for future. Choose the right mix of solutions that address the current needed as well as accommodative enough to address the future challenges. An analogy in this context could be how the civil engineers plan in building the highways. They address the current need as well as needs of future generations.

Respect the Diversity. It is every organization's dream to have a homogeneous solution used all across the partner eco system. In reality partner's technical capability and maturity vary from limited to advance. So, in making the solution choices, it is important to considerate about their abilities and priorities and make them part of the eco system.

Self-provisioning and Automated. To remove the latencies and increase the agility in the on-boarding process, partners should able to discover by themselves in an automated way. Any manual interactions included in the process would limit the ability to expand the eco system partners.

Design for network changes. Like there will be different degrees of maturity and usage of applications, the same applies for networks as well. As the generation of networks are emerging, it should consistently accommodate the network changes.

Use REST-based services. Along with technologies, standards also evolved. When compared to previous generation SOAP and UDDI directory discovery, REST based services offer a simple yet flexible mechanisms to interact. Create REST based services so that partners easily start collaborating.

Transportation and Processing Layers. Like the supply chain services are offered in a modular or micro services keep the distinction explicit between transportation and processing layers. This will help to design the implement using best in class practices.

LiveSite first. Instead of aiming for all services to complete, while building each service, design them for operations with facilities to monitor them in each phase. And also, ensure with each release stands on its own and complete from the dimension of high availability and fail safe mechanisms for automatic disaster recovery.

Abstract complexity. It is a fact that organizations build applications over a period of time and start offering individual services. To construct a workflow based solution, the underlying technology complexity need be abstract and it should appear like pick and choose the different services. Providing different REST end points make this possibility a reality.

Unify authentication and authorization.



Design for options. For every service that is intended to be published, consider the options for Post, Get, and Notify. When integrating with partners the main considerations will be based on the latency involved in the process. When the partners are getting the data, typically Notify functions are most useful ones. When the partners are interested in publishing the data, typically the Post method is used. When the partners typically want to query Get APIs come into usage.

Accountability model. Across the ecosystem, each partner is accountable for data contract, accuracy with which the information is maintained, timeliness with which data is available on demand, and also data availability for complete end-to-end business services. This is also the fundamental element in building the block chain solutions with which they collaborate.

Solution architecture. The way the supply chain partners are treated in an eco system would directly result into the number of players that are part of the system. There is a significant difference between partners treated as an external components of an internal solution versus each partner is treated as a node. When each partner is treated as a node, they recognized as an equal contributor and empowered to utilize and interact with partners of the supply chain. It is recommended to design the standards in a way that each partners plugs and plays as a node. So it is important to have a consistent and trusted mechanism so that on-boarding process is effortless and self-discoverable.

4.4.1 Establishing Integration Layers

Clear abstraction layers amongst the supply chain components makes the integration more robust and transparent. Different layers of abstraction in supply chain components are:

Partner Integration Layer. This layer contains the integration touch points when the new suppliers or partners are on-boarded. While processing a transaction, this layer is dependent on underlying ERP and processing layer.

Transaction Processing Layer. This layer contains ERP and other transactional systems. All the business processes, related transactions and different supply chain capabilities are enabled in this layer. Transactional systems, all business processes, and supply chain capabilities are enabled here to process supply chain transactions. All the services managed in this layer are called from the integration layer, but managed internally. All the system of record are maintained in this layer.



Analytics Layer: In this layer, all the transactions are pumped into a data warehouse and analytics are built. These analytics or dashboards are built to drive business managers towards an actionable outcomes. Key Performance Indicators (KPIs) are built in this layer. [17]

4.5 Rest APIs for B2B integrations

To achieve API goals, following design standards could be used in APIs and gave the same standards to partners who were establishing APIs to connect to the supply chain:

Version Number – As the services evolve and mature, they will flow and translate into the underlying APIs. Like any other software application, APIs need to contain the right version number so that partners can connect to the right version. A typical URL containing the version number of an API would look like (https://supplychainservices.Organnizationname.com/v1/orders). [17]

JSON Standards – When compared to the SOAP messages REST messages offer the simplicity and scalability. APIs that are built on REST standards typically follow the JSON standards.

OData support and handling collections – To get the data, service could expose OData format, and also OData filters can be used for hypertext. Typically these formats also extend support for pagination, sorting and filtering wherever applicable. [17]

Delta queries – Along with OData format's pagination, adding support delta queries is highly desired.

Push notifications – Typically client programs use HTTP GET when requesting for a resource. But with the advent of web 2.0, push notifications are sent automatically. This will eliminate the option of pooling. To facilitate the push notifications, webhook models are used [10]. In this model, when a push notification is sent to client, state of the current object is sent along with the notification. This will allow to notify the new changes when compared to the earlier state. [17]

Exception/error handling – It is important to have the consistent error or exception handling mechanism so that exceptions are handled in a controlled manner. In using the APIs, typically OData V4 JSON specs are followed for exception management. [17]

Azure authentication platform - If Azure active directory is implemented, services can leverage this platform and use the Azure AD token based authentication mechanism.



To ensure security, tokens expire within a configurable time, and they need to be re-generated. When a new partner is accessing the API, they need to typically generate a token and this is appended to the request header. [17]

4.6 API architecture - Supply Chain

To reduce the dependency on availability of partner service, architecture can clearly distinguish and focus on POST and GET methods. GET APIs are used for outbound or when the partner is interested in data retrieval. On the other hand, POST APIs are used for inbound requests. It means when the partners want to update the systems with the latest transaction. This approach drastically reduces the complexity of partner integration, configurations needed and minimizes the repeated trying when the partner services have a downtime.

As depicted in Fig 20, this service can divided into experience layer, repository layer and process layer.

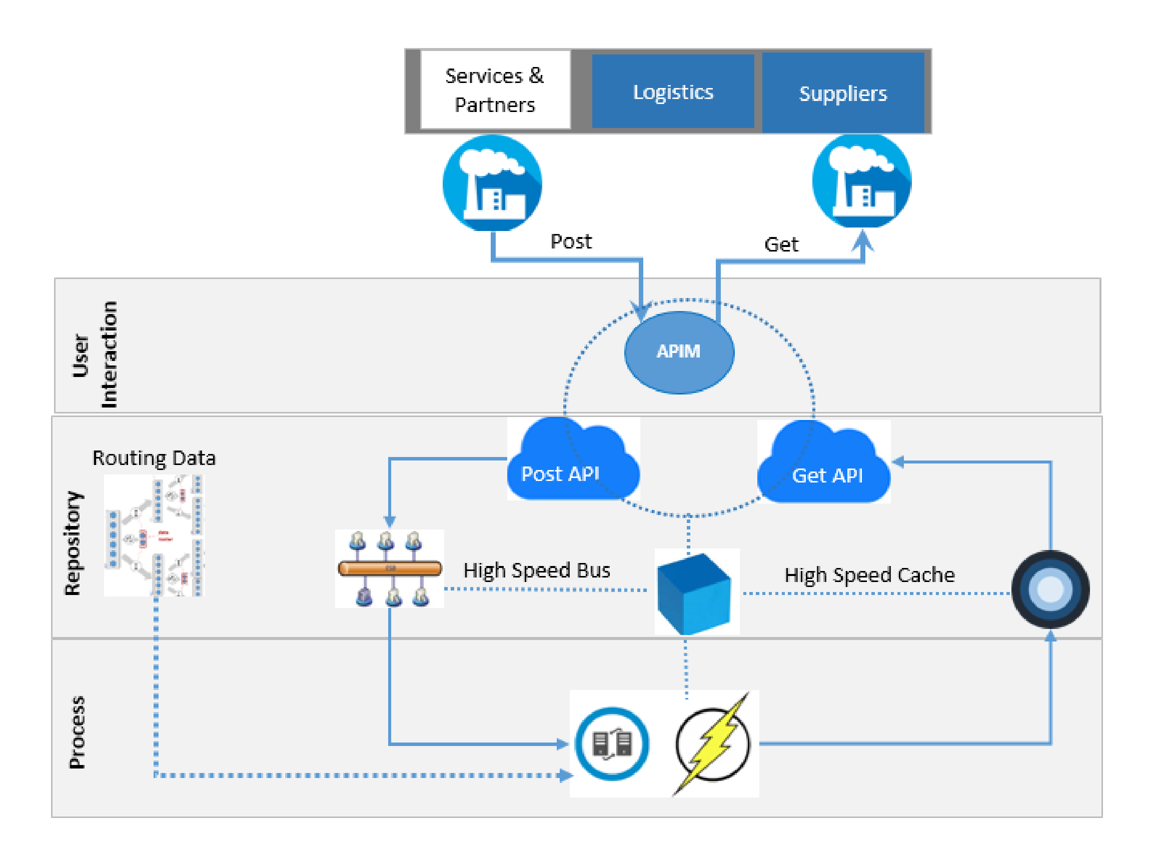


Fig 20 – API architectural view [17]



Experience layer – In this layer, the API Management gateway does most of the job. It exposes the service catalog, maintain the API contract during the interaction, and provides an easy to use experience for developer community. This layer consists of two components that work together to make the integration process for developers as simple as possible. These API management gateways can also be used for user authentication. Another important component in this layer is the REST API which is responsible for data validation and get the transaction information to the underlying repository layer. [17]

Repository layer – In this layer, there will be data persistency and the cache. This layer need to be a high performing layer. To make this layer access the different resources available on the network, native system APIs are used. [17]

Process layer - This layer acts as orchestration layer in calling the other internal APIs or applications. This is the place where the data is processed, transformed and shared across with the other applications or APIs involved. [17]

4.7 Operations and Governance

To create a truly modular, micro service-based architecture, and to keep the integrations efficient and secure, different design patterns can be used.

4.7.1 Security

As the integration amongst the supply chain partners span across the corporate boundaries, security needs becomes inherent from the inception. The common security aspects to be looked upon are authentication, authorization, message encryption and GDPR compliance.

Authentication. When different supply chain partners are joined as nodes in the network, all these end points need to be authenticated using one of the approved authentication methods. None of the nodes can afford to exist anonymously. For the integrity, all the calls amongst the partners are audited.

Authorization. Though different nodes are authenticated to be part as nodes, it should be possible to do transaction with authorized partners. This is governed using the API contract amongst the supply chain partners. All the partners respect the contract in doing the API operations.



Message encryption. To ensure data integrity, all messages should be encrypted whether data is in rest or in transit. On reaching the target node, data is decrypted and taken for processing.

GDPR compliance. All messages and APIs support meet GDPR compliance standards.

4.7.2 Monitoring

Either to get insights into the broader, end-to-end scenario or to get a detailed individual component level information, monitoring is performed. Typical techniques of monitoring include telemetry, self-healing, service health and dashboards.

Telemetry- When the transactions are getting processed through different business partners who are available as nodes in the network, it is important to have an unique identifier using which it is possible to monitor the business workflow all across the partners. These transactions are monitored using the control dashboard, and there will be mechanisms to prevent data tampering across the partners.

Self-healing- Using immutable infrastructures, all the messages support auto-healing so that message is sent to partners and received from partners in a consistent pattern.

Service health- Based on the availability, response time and other key parameters health of each service is monitored and presented back in a dashboard. When any of the services turn responding, or response times are degraded, these are automatically represented in the analytics layers. The dashboard acts like a control room in a process industry. These monitoring dashboards are set-up either for individual services or end-to-end supply chain processes.

Dashboards- Dashboards are the end results to monitor the service health.

5 API Imperative

This chapter puts focus on how the enterprises focused on re-using and cross-using its assets, How the APIs forming as gateways to reutilize the digital resources that are built over generations. This share a how the APIs form a way for reusing the digital assets of new economy, how the eco system of players enhance the values of organization assets. It also touches the typical industries which already got benefited from APIs, and a survey done at a heavy engineering manufacturing industry to assess the readiness of its suppliers to utilize the APIs.

5.1 Componentization and Inter-operability

Ever since the industrialization era, the imperative for board room managers has been to build re-usable assets to get the maximum Return On Investments (ROI) from the assets. Organizations strive to achieve not only the maximum usability from an assets, rather focused on componentizing or modularizing different elements so that innovations twine with each other like a jig-saw puzzle. In the brick and mortar era, typical assets has been machinery, warehouses, and factory units. Techniques like Toyota Manufacturing Systems, Total Preventive Maintenance (TPM) and also vast number of Operations Research techniques are deployed across the industries. Automotive industries have been pioneers in maximum component re-usability in getting the new models. Even the hi-tech industries like tele-communications where the product life is extremely short, rely on re-usable modules and engines in arriving at the new models. Indeed, the componentization has been the critical factor in mass production. Eli Whitney's rifle parts which are interchangeable give way to mass production of cards with Henry Ford's assembly lines. In the services industry, Sabre's standardized booking and ticketing process has transformed the whole Airline industry, and resulted in a collaboration that was never experienced before. And also, the payments industry completely changed the landscape with the introduction of SWIFT way of doing. This has brought huge growth in trade and commerce possible.

The same concepts manifests for digital world as well. On looking at the core offerings by digital giants like Amazon, Alibaba, Google, Microsoft, Baidu have been providing solutions that have the ability to not only solve the immediate business problems, but

they became effective launching pad for the future growth. Considering the inter-operability and modularity offered by these platforms, they not only became dominant to their immediate customers, but rather helping to build the eco system of players.

In the context of information technology industry (IT Industry), Application Programming Interfaces (APIs) are the key building blocks that provide the design modularity and interoperability. In simple terms, APIs are an architectural technique that dates back to computer science, that helps in the way to exchange information, invoke the business logics and execute transactions. In a survey conducted by Deloitte, it has observed the number of public APIs increased to 18000. Across the large enterprises globally, the private PIs could be in millions. Such a growth would enable flow of information and operations across the organizational boundaries.

5.2 Business Mandate

To take advantage of such an architecture, all the value chain partners must be able to recognize the APIs they bring, and business leaders should be willing to support and prioritize such initiatives. Only then the usage of API will move from mere “IT Department initiative” to “Business Mandate”. Given that APIs have been around for many years, and every large project in the organization has complex integrations involved, executives may tend to visualize the APIs as an incarnation of previously existing interfaces. But a vast majority of these interfaces are were designed for exchange of information between the points and are engineered to completely bespoke. These custom-built interfaces tend to be brittle and key information like customer, order, and product are often duplicated. Meanwhile, each successive project introduces new interfaces and more complexity. Value chain partners must realize that APIs were an attempt to control the chaos by encapsulating logical business concepts like core data entities (think customer or product) or transactions (for example, “place an order” or “get price”) as services. When designed efficiently, APIs could be consumed in broad and expanding ways.

Even though they present multitude of opportunities, many organizations are yet to embrace the API opportunities. In the large scale projects, developing silo integration approaches remain the rule, not the exception. Much of IT budget and effort goes to maintaining the legacy applications that were not designed to expose themselves to the value partners. Remediating that existing legacy to be API-friendly is a big feat.

5.3 Value Chain Partners Readiness - Survey

In an anonymous organization where we carried out this thesis work, APIs are acknowledged to be the strategic differentiator. But before opening up their systems and data, we conducted a survey to get the pulse of value chain partners on the awareness and readiness to participate in such an open collaboration. In this organization, currently, B2B collaboration is facilitated using EDI and those suppliers who are using EDI integrations were invited to take this survey.

To get feedback from Supplier companies, questionnaire we designed with the approval of the Supply Management Organization and it has been shared with partners to address the following dimensions:

1. **Familiarity or Awareness of APIs for B2B Collaboration**? While 56% of respondents indicated as "Yes", there is a 44% of respondents indicated as "No". Though this is in line with the hypothesis of the survey, considering these enterprises as established players with multi million revenue, this is still disappointing.

Considering the way, IT organizations are getting outsourced, the set of questions focused around the awareness among the technical team either with in or with outsourced solution provider. Central theme of these questions are around.

2. **Whether the In-house / Outsourced IT arm has the API skillset ?** While 45% of the organizations responded as "No", 55% of the respondents answered as "Yes". When compared to the earlier, this set gives a comfort that, at least the knowledge is available with the outsourced service providers.

To understand whether the organizations have gained this expertise in participating in collaborative platforms by exchanging the information or these APIs are developed for internal consumption, next set of questions are developed to understand this dimension

3. **Does the organization developed any proof-of-concepts on APIs? -** Surprisingly 78% of the organizations responded they have not developed on concept models. Only 22% of the respondents answered with an emphatic "Yes".

This brings out fact that, many organizations are still continuing to reinvest in improving and maintaining the bespoke interfaces that are developed across the systems, than exploring the possibilities of exposing the business logic and data



To investigate further in which areas these APIs are developed, investigation is further continued to ask

4. **Whether these APIs are developed for External / Internal usage?** 100% of the respondents answered for this one as "External". This is a very encouraging result. These are certainly the partners who recognized the value of opening the organization's data and business logic in a controlled way, yet recognize the benefits of the same. These partners come out as Early Adaptors and stay ahead of other participants

At this stage, the attention went further to identify in which functional areas the partners see more value in. So, the partners are asked whether they could share this information.

5. **In which Operational areas, these prototypes are built in?** Respondents came back with two distinct areas as Sales Operations and Procurement Operations

Information shared in procurement operations can further classified into "Purchase Orders" for immediate time frame (2-8 weeks out), Unconfirmed Purchase Orders (8-18 weeks out) and there after the Purchase Forecast for longer term. Organization feed this information further into the Supplier's planning systems. This will help in reducing the Work in Progress inventory and also the Advance Shipment Notifications in a synchronized way to the manufacturer

In a manufacturing industry, both these functional areas as highly suitable for external collaboration – one on the sales side and next one on purchase side.

To identify whether any of these prototypes in these areas are resulted in production usage, information is further seek as

6. **Whether their organization is already using APIs for B2B integrations?** 100% of the respondents came back with an assertive "No". This reflects the fact that, usage of APIs is still not included as one of the Strategic Priorities of the organization (or to say that not part of Business Strategy). As long as organizations consider these as integration technologies rather than Strategic Enablers, this situation may continue.

To understand whether the above inference matches with the reality (not to make any assumption), next set of questions are designed around.



7. **Whether the Business Users are well aware of advantages from APIs?**

   11% of respondents reports they are “Well Familiar”, 22% of the respondents reported they are “Familiar” and, 67% of the respondents reported as they are “Not Familiar”

In this case, the results are self-explanatory on what to expect further from the organizations. While the first 33% start paving the way towards opening up multiple channels and start creating new business models, the organizations in the 67% bucket will continue to grind the existing mechanisms to get max out of them.

Though in the current state, one may or may not be familiar with leveraging the technology trends, if a good forerunner shows the way, many organizations will be interested to follow and enjoy the benefits of such collaboration. To understand the willingness of the enterprises to follow such path, it was further explored on.

8. **If this organization (anonymous organization where the study is conducted) creates APIs for B2B message exchange, will your organization be willing to participate in testing?** 78% of the respondents came back with and emphatic “Yes” and, 22% chose to answer as “No”

This clearly demonstrates the hypotheses with which the survey is conducted.

When participants are asked to indicate the functional areas where they are willing participate / wish to see the APIs available, the areas where the interest has been expressed in the order of more votes goes as Purchase Order (PO) issuance and order confirmation, pick up request and dispatch of goods, component traceability, inventory visibility and, forecast collaboration.

In conclusion, it is clearly evident that, when leaders of a specific value chain takes the initiative to create the collaborative platforms, the other players follow. Executives need to recognize this strategic priority, include as part of Business Strategy and create new opportunities by opening up the new channels of delivery / create new business models. In the coming years, the number of APIs would see an exponential growth akin to the components in manufacturing industry.



6 Conclusions and Research findings

In this chapter the research findings are presented starting from the current challenges and preparing towards the future and explaining on how API way of integrations could lead to a sufficient part of B2B integrations. What could be the perceivance currently in the industry and who can influence it to overcome the challenges foreseen.

6.1 Analysing the Current Challenges and Preparing for Future

The traditional solution in supply chain for integrating suppliers has its own advantages as it is an accepted convention that already follows business processes but also has the lacking because of which the growing business needs cannot be catered efficiently. APIs would be a good consideration in this context and could solve the following main challenges with the traditional integration methods:

On-boarding new partners is too slow and expensive. Adding a new supplier to the supply chain is a unique, time- and labor-intensive process. Code mapping needs to be developed for each supplier integration. A team of program managers and developers was typically assigned to each new integration, which involved coordination, connectivity testing, and end-to-end testing, all of which would take long time to complete.

If the management and maintenance of the integrations are not done in-house by the organization's own IT, then there is no reliability in end-to-end visibility. The visibility into the supply chain process was limited and as a result, end-to-end supply chain picture is incomplete.

Operation costs are high. The costs to run infrastructure, upgrades, and maintenance are too high if the integrations are maintained with the Organization's IT department. In such case there could be chance of inability to scale the infrastructure to meet the demand and variable growth.

If the operations and maintenance is outsourced, both the implementation and maintenance costs are very high.

Hence looking at the above, the high-level goal of the new solution should enable digital transformation taking care all the advantages of the current methods and try to cover the lacking. We needed our supply chain platform to move at the speed of our business and enable cost savings and improvements, rather than hold us back. Following could be short list of goals for the future solutions:



Accelerate development. A shorter development cycle so that new features and fixes could be integrated faster.

Build micro services based architecture. It has been identified that micro services is the most appropriate architectural model for achieving a modular, yet fully integrated, solution. The more we individualized components and created them to interact cohesively with the whole, the easier it would be to on-board new partners.

Achieve highly elastic scalability. There is a need of rapid elasticity. The nature of the online, retail business means that high-volume buying periods—like holidays—can escalate quickly in unpredictable patterns. The solution needs to scale automatically and effortlessly to meet demand.

Reduce infrastructure and maintenance costs depending on whether it is in-house or outsourced.

Improve business visibility. New solution had to give an end-to-end picture of our business, so that accurate and effective ways are found to improve and grow the supply chain.

Increased availability and disaster recovery. High availability is required for global supply chain integrations and at the same time, reduce the administrative overhead and improve the speed and automation of disaster recovery.

6.2 APIs as Solution for Future Integrations

APIs (Application Programming Interface) were created as a means to make data and applications available as a service to a broad base of consumers. When Amazon launched S3 (Simple Storage Service) in 2006, and made the service available via API, it opened the door for entire enterprises to be supported through the cloud. While organizations want to reduce cost barriers to integration and use data as a competitive advantage, APIs offer the promise of lower integration costs, the ability to make data more accessible, and create new streams of revenue.

API is a powerful technology primarily created for synchronizing web applications, mobile apps, etc. In simple terms, API provides interfaces that allow us to access data in systems in real time. There is no doubt that with companies adopting APIs business to business integration will become easier and message tracking efficient owing to its analytics capabilities but it’s too early to say EDI will be a thing of the past. At this stage, APIs can only replace a subset of EDI transactions and it will probably take about another decade

to see the retirement of EDI that too when all organizations unanimously decide to change the business processes at their end to adopt APIs or any similar technology that will crop up in future as the mode of B2B communication.

Implementing APIs for B2B integrations has more benefits when opening a business process than you may think. Numerous studies are predicting a considerable increase in the number of companies that decide to integrate an API to improve their B2B strategies (business-to-business*)*. New data sources, either from new partners or from third parties, can be integrated using an API. The API makes managing and organizing these sources more agile, because it can also work on the management systems and processes of the inherited data. It will reduce security problems in the development processes, as the implementation of an API will mean your company's developers will have to compartmentalize the business applications and processes and reinforce the most sensitive internal systems. The architects and developers in your company must assess the solutions available to ensure the apps' performance so they can take them into account in the strategy. To improve this work process, the net must be optimized and integrated.

APIs will allow companies to combine software as a service and mobile apps with more traditional B2B protocols, and also with the standards laid down in the different regulations.

6.3 How the Change Could be Perceived – Conclusions

Restful APIs for B2B integrations are being looked upon for B2B integrations. But, it is good to analyze that are the suppliers in the ecosystem willing to move towards API from the traditional systems. Previous investments in legacy technology might mean it will be extremely unlikely any business can communicate exclusively via API. Gartner estimates that 25% of B2B interactions will be performed through APIs by 2020 while the majority will still be handled by “legacy approaches”. Organizations looking to take advantage of APIs with trading partners should consider whether their technology is able to support both approaches effectively. Organizations looking to replace older integration methods with APIs for their B2B transactions should also consider the level of asymmetry, and whether their trading partner will be willing to conform data to the intended business process. In those cases the information might need be translated somehow. The integration outsourcing companies might need to be consulted well before planning the change.



As presented in Chapter – 5 about the survey conducted with the suppliers who are already using that traditional way of integration with the anonymous organization where the research has been conducted, the results looks like this. More than 50% of the organizations are aware of API for B2B collaboration but none of them are using this method of B2B integration. But, more than 70% are interested to participate in the prototype testing process in order to increase their awareness.

APIs, Supply chain integrations context, would require more collaboration. Each trading partner must agree on the semantics and granularity of the data. For example, one trading partner may refer to a data set as the “ship to location” and the other partner the “shipping address”. The documentation for each transaction should be clear enough to take care of these issues which might arise. Also, a single transaction of purchase order may require 50 operations within an API. If the message is split into a series of calls, the consuming back-end should made sure to receive that information without performance issues.

This new API based solution would create a simpler, faster, and more dynamic supply chain management process. It would be easier for partners to onboard, interact, and conduct business with and the organization would have end-to-end control and visibility of processes. The solution would allow to focus not only on maximizing revenue, but on doing so optimally and sustainably by building on a strong services-based foundation across entire supply chain, transforming the way we do business. This transformation can be taken further by exploring technologies like Blockchain, Internet of Things, and cognitive bots [9]. These investments would give a competitive edge in the supply chain and bring maximum value to Suppliers, partners and customers.

Benefits of API for B2B integrations: With the study conducted, the following benefits can be perceived : Low management overhead, Standard integration architecture, Scalable architecture, Global platform for bi-directional data sharing, An agile and flexible platform, Low cost.

On a closing note, APIs could be a good part of the future B2B integrations especially in Supply chain. As introduction of any new technology or integration method does not go very easily especially in the case of Supply chain where the transactions are tied with several partners together, this transition might not be very smooth. Bigger organizations who has the good IT infrastructure should start with it and try convincing the partners and suppliers and partners and bring awareness and develop some standards to take this forward.



References

1. NetSuite.com. Do you know what ERP is? Learn how ERP can help your business with this informative article. Business Software, Business Management Software – NetSuite. [cited 2018Nov21]. http://www.netsuite.com/portal/resource/articles/erp/what-is-erp.shtml

2. Gillai B, Yu T. B2B Integration Business Value and Adoption Trends. Open Text, Stanford Business Global Supply Chian Management Forum; 2013. https://www.opentext.ca/file_source/OpenText/en_US/PDF/opentext-whitepaper-information-exchange-b2b-integration-en.pdf

3. B2B Integration. What is B2B Integration? « B2B Integration. [cited 2018Nov21]. http://www.b2bintegration.co.uk/what-is-b2b-integration/

4. Forrester: Marketing : Forrester Forecasts US Online Retail To Top $500B By 2020. [cited 2018Nov21]. https://www.forrester.com/report/The Future Of EDI/-/E-RES58057#

5. Robinson S. Beyond B2B, A Digital Reality Check. Three technology drivers which will impact you in 2018. [cited 2018Nov21]. http://www.b2b.com/blog/2014/06/20/b2b-integration-evolving/

6. Ghosh A. 7 Reasons Why the API Trend Will Stay with B2B Marketers in 2018 and Beyond. MarTech Advisor. [cited 2018Nov21]. https://www.martechadvisor.com/articles/cloud-amp-integration/7-reasons-why-the-api-trend-will-stay-with-b2b-marketers-in-2018-and-beyond/

7. History of APIs. History of APIs. [cited 2018Nov21]. http://history.apievangelist.com/

8. API Provider for API Consumer: What to Have on Your Documentation Page. API2Cart - Unified Shopping Cart Data Interface. 2017 [cited 2018Nov21]. https://api2cart.com/api-technology/api-provider-api-consumer-documentation-page/

9. Weldon M., The future X network a Bell Labs perspective. CRC press. 2015.

10. API for Partner Integrations. GoodHire. [cited 2018Nov23]. https://www.goodhire.com/partners/api-documentation#getting-started

11. What is an API Consumer? TCP states - explained - Apigee Community. [cited 2018Nov23]. https://community.apigee.com/questions/43660/what-is-an-api-consumer.html.

12. Whitepapers. MuleSoft. [cited 2018Nov11]. https://www.mulesoft.com/whitepaper/integration-use-cases

13. SMC³. Pros and Cons of Supply Chain Transactional APIs versus EDI Logistics Communication Technologies - Supply Chain 24/7. Supply Chain 24 7. [cited 2018Nov11]. http://www.supplychain247.com/article/apis_versus_edi_technology



14. Business Matters. APIs & their role in improving the customer experience. Business Matters. Business Matters; 2018 [cited 2018Nov11]. https://www.bmmagazine.co.uk/in-business/apis-their-role-in-improving-the-customer-experience/

15. Neill MO, Golluscio E. Master B2B Ecosystem Integration With a Blended EDI and API approach. Master B2B eco system. Gartner; 2017 [cited 2018Nov11]. http://www.cleo.com/content/uploads/Gartner-Master-B2B-Ecosystem-Integration-With-a-Blended-EDI-and-API-Approach.pdf

16. Gillai B, Yu T. B2B Integration: Business Value and Adoption Trends. B2B Integration: Business Value and Adoption Trends. Stanford Business Graduate School; 2013 [cited 2018Nov11]. https://www.gsb.stanford.edu/sites/gsb/files/publication-pdf/b2b-integration.pdf

17. Creating an integrated plug-and-play supply chain with serverless computing. Software Asset Management – Microsoft SAM. [cited 2018Nov11]. https://www.microsoft.com/itshowcase/Article/Content/973/Creating-an-integrated-plugandplay-supply-chain-with-serverless-computing-supply-chain-with-serverless-computing

